UNIVERSITAT POMPEU FABRA

Comparative Evaluation and Combination of Automatic Rhythm Description Systems

José Ricardo Zapata González

TESI DOCTORAL UPF / 2013

Director de la tesi:

Dr. Emilia Gómez
Dept. of Information and Communication Technologies
Universitat Pompeu Fabra, Barcelona, Spain

Dissertation submitted to the Department of Information and Communication Technologies of Universitat Pompeu Fabra in partial fulfillment of the requirements for the degree of

DOCTOR PER LA UNIVERSITAT POMPEU FABRA,

with the mention of European Doctor.

Music Technology Group (http://mtg.upf.edu), Dept. of Information and Communication Technologies (http://www.upf.edu/dtic), Universitat Pompeu Fabra (http://www.upf.edu), Barcelona, Spain.

A Mi Familia, lo mas importante en mi vida

Mi Madre, por su tenacidad
Mary, por el amor a la familia
Caro, por estar siempre conmigo
y ser mi fiel amiga
Daniel, por su energía
Isabel C, por su creatividad
July, por su ternura
y especialmente
a mi Padre
Por su amor, constante dedicación y
por ser el mejor modelo a seguir.

Acknowledgements

"Happiness only real when shared"
- Into the wild -

The first time when i attempt to engage the beat tracking problem was 8 years ago as part of my master's thesis in Colombia. In that moment beat trackers were in their early stages of being applied as a feasible solution. I try in that moment to improve the code that i had found on Internet, but that attempt only left me with a certain uneasiness and questions about the topic. I lacked then the supporting team necessary to carry this out. I was fortunate enough to find this wonderful and supportive group of individuals at the MTG in Barcelona and SMC in Porto. Right now finishing my Phd and after living four years in the great city of Barcelona, I realize that i did it and I could make all this possible because the support of wonderful people. Among those who have made this possible, i would like to thank:
First, I would like to thank my supervisor, Emilia Gómez. For the time that she spent talking to me about my work, for giving me the confidence and support in all stages of the Phd process and commenting on the numerous drafts of the conference papers and this document. Thanks to Xavier Serra, for giving me the opportunity to join the MTG and being a role model as a leader and as a person. Special thanks to the "beat tracking team" in Porto, who have been without a doubt a very important part in this thesis. Thank you for the friendship and special thanks to Matthew E.P. Davies, for the discussions, answering my questions and helping me a lot with the beat tracking code; Andre Holzapfel, for the discussions and the hospitality; Fabien Gouyon, for the comments and giving me the opportunity to join his research group in Porto; João Lobato, for the hospitality.
In timeline, thanks to JP carrascal (for the guitar jams, the creativity, conversations and the support in BCN) , Mohamed Sordo (for being a good friend and bringing me to your family and life in Morocco), Sergio Giraldo (for the conversations, guitar jams, salsa dances and being very good friend). Thanks to my Phd mates, Agustin Martorell, Justin Salamon (thanks for proofreading of my papers), Saso Musevic for the discussions and Dmitry Bogdanov (thanks, for the beat tracker implementation). To my classmates in SMC 2009-2010, Especially those who are still in MTG, Frederic Font, Marco Marchini, Panos Papiotis (thanks for lending me your guitar in the compilation times), Alvaro Sarasua and Marti Umbert. There are many more people from the MTG that I would like to acknowledge, these are: Eduard Aylon, Perfecto Herrera, Jordi Funollet, Piotr Holonowicz, Ricard Marxer, Oscar Mayor, Hendrik Purwins, Nicolas Wack, Jordi Bonada, Jordi Janer, Juan Jose Bosch,

Gopala Krishna Koduri, Sebastián Mealla, Alastair Porter, Rafael Ramirez, Sertan Sentürk, Ajay Srinivasamurthy, Zacharias Vamvakousis, Alfonso Perez, Joan Serrà, Carles F. Julià, Daniel Gallardo, Gerard Roma, Graham Coleman and Mathiu Bosi. (sorry if I am forgetting someone!). To the MTG administration staff, who always helped me, Cristina Garrido, Alba Rosado and Sonia Espí thank you a lot. To Lydia Garcia and Vanessa Jimenez for helping with all the paperwork in the university.
Thanks to all the great people that I met these years, special thanks to my flatmates and guests in "Hotel Diagonal 104" for all the good times and the guitar parties at home: Tomas and Natalia (for their friendship), Lina (for being my little sister), Carlos (for the good meals), Robin (for proofreading my papers and the good times), Lorena (for all the talks), Ana (all the talks about music and movies), Sankalp (Thanks for all the good meals, the conversations, the spontaneous music jams and to let me treat you as my brother) and Paula B (for the all the conversations, music jams, the good meals, salsa dances and all the support in the last stage of this thesis).
I would also like to acknowledge Juan David Gómez (my big brother), Carolina Zapata (my beloved sister) for their patience of proofreading this thesis. Special mention to Paula Betancur, Sankalp Gulati and Ajay Srinivasamurthy for all the help in the preparation of the Phd defense slides thank you.
Thanks to Fabien Gouyon, Juan P. Bello and Xavier Serra, the defense board of the thesis for all their constructive comments and their patience reading the thesis.
Thanks to the good people with whom I am very grateful to share very nice time in Barcelona and Europe, Sebastian V, Sergio B, Antonio E, Melisa E, La vecindad (Camilo M, Clara L, Wen C, Maria I. U, Silvana P), Veronica R, Juana DLC, Andres B, Ariana P, Ermengol S, Estefania C, Marius M, Soledad R, Luisa C.
Last, but not least, I want to mention my friends in Colombia, their past and present support made it possible for me to reach this point: Rafael Piedrahita, Juan Pablo Cadavid, Diego Muñoz, Leonardo Posada, Federico G, Isabel G, Diana U, Camilo Arango, Carlos Mario Guarin, Lina Acevedo, Maria Clara Juanita de la Cuesta, Jorge Alberto Ramirez, Mari Hoyos, Tomas White. My friends and colleagues in the UPB Medellin: Roberto Hincapie, Javier Serra, Cristina Gomez, Claudia Carmona, Luciano Gallon and Jackson Reina.
Finally and most importantly, my dad, Juan Carlos; my moms, Doris and Mariela; my sisters Carolina, Isabel Cristina, Juliana; my brother Daniel, my new brother Fabrizio; and finally my family in EEUU and Colombia. Big thanks to my whole family who have always supported me and believed in me.

This research has been carried out while i was a part of the Music Technology Group at Universitat Pompeu Fabra (UPF) in Barcelona, Spain from Sep. 2009 to Sep. 2013 and from May. 2011 to Jul. 2011 and Apr 2012, at the SMC at INESC in Porto, Portugal. I received economic support from the R+I+D scholarship, from el Departamento Administrativo de Ciencia, Tecnología e Innovación en Colombia (COLCIENCIAS) and Universidad Pontificia Bolivariana (UPB) Medellin, Colombia. The conference support provided by the projects of the spanish ministry of science and innovation DRIMS (MICINN - TIN2009-14247-C02-01.), SIGMUS (MINECO-TIN2012-36650.) and Mires (EC-7PM-MIReS).

Abstract

The automatic analysis of musical rhythm from audio, and more specifically tempo and beat tracking, is one of the fundamental open research problems in Music Information Retrieval (MIR) research. Automatic beat tracking is a valuable tool for the solution of other MIR problems, because enables the beat-synchronous analysis of music for tasks such as: structural segmentation, chord detection, music similarity, cover song detection, automatic remixing and interactive music systems. Even though automatic rhythm description is a relatively mature research topic in MIR and various algorithms have been proposed, tempo estimation and beat tracking remain an unsolved problem. Recent comparative studies of automatic rhythm description systems suggest that there has been little improvement in the state of the art over the last few years. In this thesis, we describe a new method for the extraction of beat times with a confidence value from music audio, based on the measurement of mutual agreement between a committee of beat tracking systems. Additionally, we present an open source approach which only requires a single beat tracking model and uses multiple onset detection functions for the mutual agreement. The method can also be used to identify music samples that are challenging for beat tracking without the need for ground truth annotations. Using the proposed method, we compiled a new dataset that consist of pieces that are difficult for state-of-the-art beat tracking algorithms. Through an international evaluation framework we show that our method yields the highest AMLc and AMLt accuracies obtained in this evaluation to date. Moreover, we compare our method to 20 reference systems using the largest existing annotated dataset for beat tracking and show that it outperforms all of the other systems under all the evaluation criteria used. In the thesis we also conduct an extensive comparative evaluation and combination of automatic rhythm description systems. We evaluated 32 tempo estimation and 16 beat tracking state-of-the-art systems in order to identify their characteristics and investigated how they can be combined to improve performance. Finally, we proposed and evaluated the use of voice suppression algorithms in music signals with predominant vocals in order to improve the performance of existing beat tracking methods.

Resumen

El análisis automático musical del ritmo en audio, y más concretamente el tempo y la detección de beats (Beat tracking), es uno de los problemas fundamentales en recuperación de información de Musical (MIR). La detección automática de beat es una valiosa herramienta para la solución de otros problemas de MIR, ya que permite el análisis sincronizado de la música con los beats para tareas tales como: segmentación estructural de musica, detección de acordes, similitud músical, la detección de versiones de una cancion, mezcla automática de canciones y sistemas interactivos músicales. Aunque la descripción automática de ritmo es un area de investigación relativamente madura en MIR y diversos algoritmos se han propuesto, la estimación tempo y la detección de beats siguen siendo un problema sin resolver. Recientes estudios comparativos de estos sitemas sugieren que ha habido pocas mejoras en el estado del arte en los últimos años. En esta tesis, describimos un nuevo método para la extracción de beats en señales de audio que mide el grado de confianza de la estimación, basado en la medición del grado de similitud entre un comité de sistemas de detección de beats. Además, se presenta una variante a este metodo que sólo requiere de un modelo único de detección de beats y que utiliza varias funciones de detección de onsets como comité para la estimación de similitud. Estos métodos se puede utilizar también para identificar canciones que son difíciles para la detección de beats sin la necesidad de intervención humana. Utilizando el método propuesto, Hemos compilado una nueva base de datos que se compone de piezas que son difíciles para los algoritmos de detección de beats. A través de una evaluación internacional se demuestra que nuestro método proporciona el más alto resultado en las medidas de ALMc y AMLt obtenidas en esta evaluación hasta la fecha. Además, comparamos nuestro método con 20 sistemas de referencia en la mas grande base de datos existente para la detección de beats y demostramos que supera a todos los otros sistemas en todos los criterios de evaluación utilizados. En este trabajo también llevamos a cabo una extensa evaluación comparativa de los sistemas actuales de descripción automática ritmo. Para esto, Evaluamos 32 algoritmos de tempo y 16 sistemas de detección de beats que reflejan el estado del arte en el area con el fin de identificar sus características e investigar la forma en que se pueden combinar para mejorar el rendimiento en esta area. Por último, proponemos y evaluamos el uso de algoritmos de supresión de voz para señales de música con voz predominante con el fin de mejorar el rendimiento de los métodos de detección de beats.

Contents

List of figures

List of tables

List of abbreviations

Abbreviation	Description
ACF	Autocorrelation function
BAS	Bandwise accent signal
BF	Bank-comb filter
BEF	Beat emphasis function
BPM	Beats per minute
BT	Beat Tracker
BT-MMA$_D$	Beat tracker - Mean Mutual Agreement using information gain
CFB	Comb filter bank
CSD	Complex spectral difference
DFT	Discrete Fourier transform
DP	Dynamic programming
EF	Energy flux
FFT	Fast fourier transform
HF	Harmonic feature
HMM	Hidden Markov model
InfGain	Information gain measure
IOI	Inter onset interval
MA	Multiple agent system
MAF	Mel auditory feature
MGP	Mean ground truth performance
MIR	Music information retrieval
MIREX	Music information retrieval evaluation exchange
MMA	Mean mutual agreement
MaxMA	Maximum mutual agreement
MinMA	Minimum mutual agreement
ODF	Onset detection function
SFX	Spectral flux
SP	Spectral product
STFT	Short-time Fourier transform
TAP-MGP	Mean performance of the taps compared to ground truth
TAP-MMA	Mean mutual agreement taps

CHAPTER 1

Introduction

"If I have seen further it is by standing on the shoulders of giants"
- Sir Isaac Newton -

1.1. Motivation

Rhythm, along with harmony, melody and timbre, are one of the most fundamental aspects of music, sound, by its very nature, is temporal while the word rhythm, in its most generic sense, is used to refer to all of the temporal aspects of a musical work, whether it is represented in a score, measured from a performance, or existing only in the perception of the listener. In order to build a computer system capable of intelligently processing music, it is essential to design representation formats and processing algorithms for the rhythmic content of music (Gouyon & Dixon, 2005).

The content analysis of musical audio signals has received increasing attention from the research community, specifically in the field of music information retrieval (MIR) (Pampalk, 2006). MIR aims to retrieve musical pieces by processing not only text information, such as artist name, song title or music genre, but also by processing musical content directly in order to retrieve a piece based on its rhythm or melody (Typke et al., 2005). Since the earliest audio beat tracking systems by Dixon (1997); Goto & Muraoka (1994); Scheirer (1997) in the mid to late 1990s, there has been a steady growth in the variety of approaches developed and the applications to which these beat tracking systems have been applied. The use of automatic rhythm description has become a standard tool for solving other MIR problems, e.g. structural segmentation (Levy & Sandler, 2008), chord detection (Mauch et al., 2009), music similarity (Holzapfel & Stylianou, 2010), cover song detection (Ravuri & Ellis, 2010), automatic remixing (Hockman et al., 2008) and interactive music systems (Robertson & Plumbley, 2007); by enabling "beat-synchronous" analysis of music.

While many different beat tracking and tempo estimation techniques have been proposed over the last five years, e.g. for beat tracking (Böck & Schedl,

2011; Davies & Plumbley, 2007; Degara et al., 2012; Dixon, 2007; Ellis, 2007; Peeters, 2009) and for tempo estimation (Gainza & Coyle, 2011; Gkiokas et al., 2010; Peeters, 2010), recent comparative studies of rhythm description systems suggest that there has been little improvement in the state of the art over the last seven years (McKinney et al., 2007) and the method by Klapuri et al. (2006) is still widely considered to represent the state of the art for both tasks. Current approaches for rhythm description focus on the analysis of mainstream popular music with clear and stable rhythm and percussion instruments, which facilitates this tasks. These approaches mainly consider periodicity of intensity descriptors (principally onset detection functions) to locate the beats, and then to estimate the tempo. Nevertheless, they usually fail when they are analyzing other music genres like classical music, because this type of music presents tempo variations; in other words, it does not include clear percussive and repetitive events. The same problem appears with acapella or choral music (only singing voice with a fixed and evident periodic rate), acoustic music, some jazz and pop music (Gouyon & Dixon, 2005).
While the efficacy of automatic rhythm description systems can be evaluated in terms of their success of these end-applications, e.g. by measuring chord detection accuracy, considerable attention has been given to the beat tracking the evaluation of through the use of annotated test databases in particular the MIR community has made a considerable effort to standardize evaluations of MIR systems. As part of this effort, there are specific tasks in the Music Information Retrieval Evaluation eXchange (MIREX)(Downie, 2008) initiative to evaluate audio beat tracking and tempo induction systems. In the small number of comparative studies of automatic beat tracking algorithms with human tappers (Collins, 2006; Davies & Plumbley, 2007; Holzapfel et al., 2012b; McKinney et al., 2007; Scheirer, 1998) musically trained individuals are generally shown to be more adept at tapping the beat than the best computational systems. Given this gap between human performance and computational beat trackers, we consider that beat tracking is not yet a solved problem.

1.2. Definitions

Musical rhythm is used to refer to the temporal aspects of a musical work and its components are Beat, Tempo, Meter, Timing and Grouping presented in Figure 1.1. For the sake of understanding the computational approaches of automatic rhythm description methods it is important to emphasize the difference between musical pulse and beat. It is often, incorrectly assumed that the musical pulse which can be felt by a human being corresponds to a beat. To address that we have focused our research on automatically estimate the musical beats in audio signals (Beat tracking) and tempo estimation related to the beats per minute in a song.

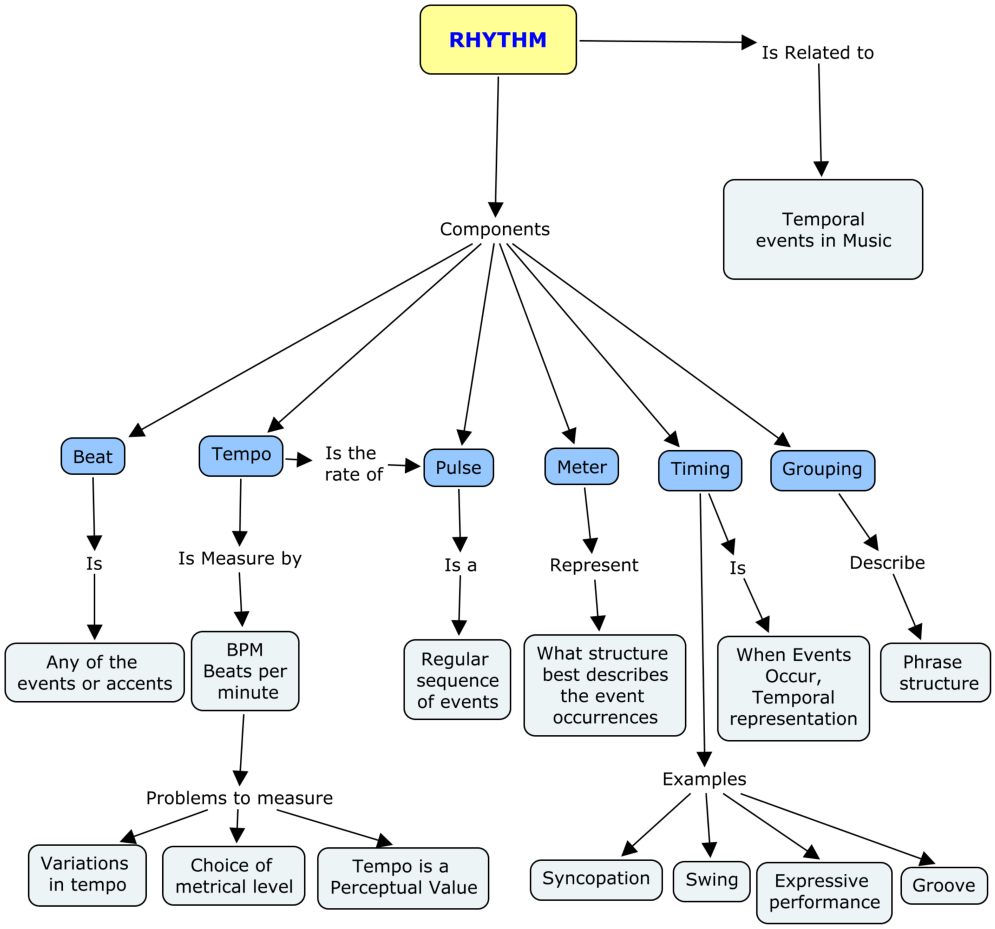


Figure 1.1: Rhythm Components

1.2.1. Pulse

Musical pulse is defined by Cooper & Meyer (1960) as:

> *"One of a series of regularly recurring, precisely equivalent stimuli ... Pulses mark off equal units in the temporal continuum."*

In a perceptual way, Berry (1987) defined pulse as:

> *"The felt, underlying, at times regularly recurrent unit by which music's times span is measured and its division felt at some specific level."*

Commonly, "pulse" and "beat" are often used indistinctly and refer both to one element in such a series and to the whole series itself. It is not always correct to

assume that the pulse indicated in a score (Maelzel Metronome) corresponds to the "foot-tapping" rate, nor to the actual "physical tempo" that would be an inherent property of audio streams (Drake et al., 1999).

1.2.2. Beat

Beat is defined by Berry (1987) as:

> *"The basic unit of time, the pulse of the mensural level."*

Additionally, Handel (1989) describes the relation between beat and pulse as:

> *"Typically what listeners entrain to as they tap their foot or dance along with a piece of music."*

The beat perception is an active area of research in music cognition, in which there has long been an interest in the cues listeners use to extract a beat. Temperley & Bartlette (2002), list six factors that most researchers agree are important in beat finding (i.e., in inferring the beat from a piece of music). These factors can be expressed as preferences:

1. for beats to coincide with note onsets.
2. for beats to coincide with longer notes.
3. for regularity of beats.
4. for beats to align with the beginning of musical phrases.
5. for beats to align with points of harmonic change.
6. for beats to align with the onsets of repeating melodic patterns.

1.2.3. Tempo

Tempo, is defined as the number of beats in a time unit (usually the minute). There is usually a preferred pulse, which corresponds to the rate at which most people would tap or clap in time with the music. However, the perception of tempo exhibits a degree of variability. Differences in human perception of tempo depend on age, musical training, music preferences and general listening context (Lapidaki, 1996). They are, nevertheless, far from random and most often correspond to a focus on a different metrical level and are quantifiable as simple ratios (e.g. 2, 3, 1/2 or 1/3) (Polotti, 2008). In this work, the automatic tempo estimation is related to detect the beats per minute.

1.2.4. Metrical levels

Lerdahl & Jackendoff (1983) Generative Theory of Tonal Music (GTTM) define Meter as the metrical structure of a musical piece based on the coexistence of several pulses (or "metrical levels"), from low levels (small time divisions) to high levels (longer time divisions). The segmentation of time by a given low-level pulse provides the basic time span to measure music event accentuation whose periodic recurrences define other higher metrical levels.
GTTM also formalizes the "musical grammar", the distinction between grouping structure (phrasing), and metrical structure by defining rules. Whereas the grouping structure deals with time spans (durations), the metrical structure deals with duration-less points in time-beats that obey the following rules. First, beats must be equally spaced. A division according to a specific duration corresponds to a metrical level. Several levels coexist, from low levels (small time divisions) to high levels (longer time divisions). There must be a beat of the metrical structure for every note in a musical sequence. A beat at a high level must also be a beat at each lower level. At any metrical level, a beat that is also a beat at the next higher level is called a downbeat, and other beats are called upbeats.
The metrical levels can be divided into three hierarchical levels: *Tatum*, *Tactus* (Beat), *Bar* or measure. The relations between the audio signal and the metrical levels are represented in Figure 1.2 using a representation of an audio excerpt of a percussive performance of samba rhythm. The sequence of note onsets, related with each drum hit of the audio is shown in Figure 1.2(b). The *tatum*, the low metrical level, is defined by Bilmes (1993) as the shortest commonly time interval. The *tactus* or beat, Figure 1.2(d), is defined by Lerdahl & Jackendoff (1983, p.21) as the preferred human tapping tempo and the computational approach of this task is called *beat tracking*. The bar, Figure 1.2(e), is the highest metrical level and is typically related to the harmonic change rate or to the length of a rhythmic pattern.

1.2.5. Automatic rhythm description

The aim of automatic rhythm description is parsing acoustic events that occur in time into more abstract notions of tempo, timing and meter. Algorithms described in the literature differ in their goals, some of them derive beats and tempo of a single metrical level, others try to derive the complete transcription (i.e. musical scores), others aim to determine some timing features from musical performances (such as tempo changes, event shifts or swing factors), others focus on the classification of music signals by their overall rhythmic similarities, while others look for rhythm patterns. Nevertheless, these computer programs share some functional aspects (feature list creation, pulse induction, Figure 1.3), as pointed out by Gouyon & Dixon (2005).

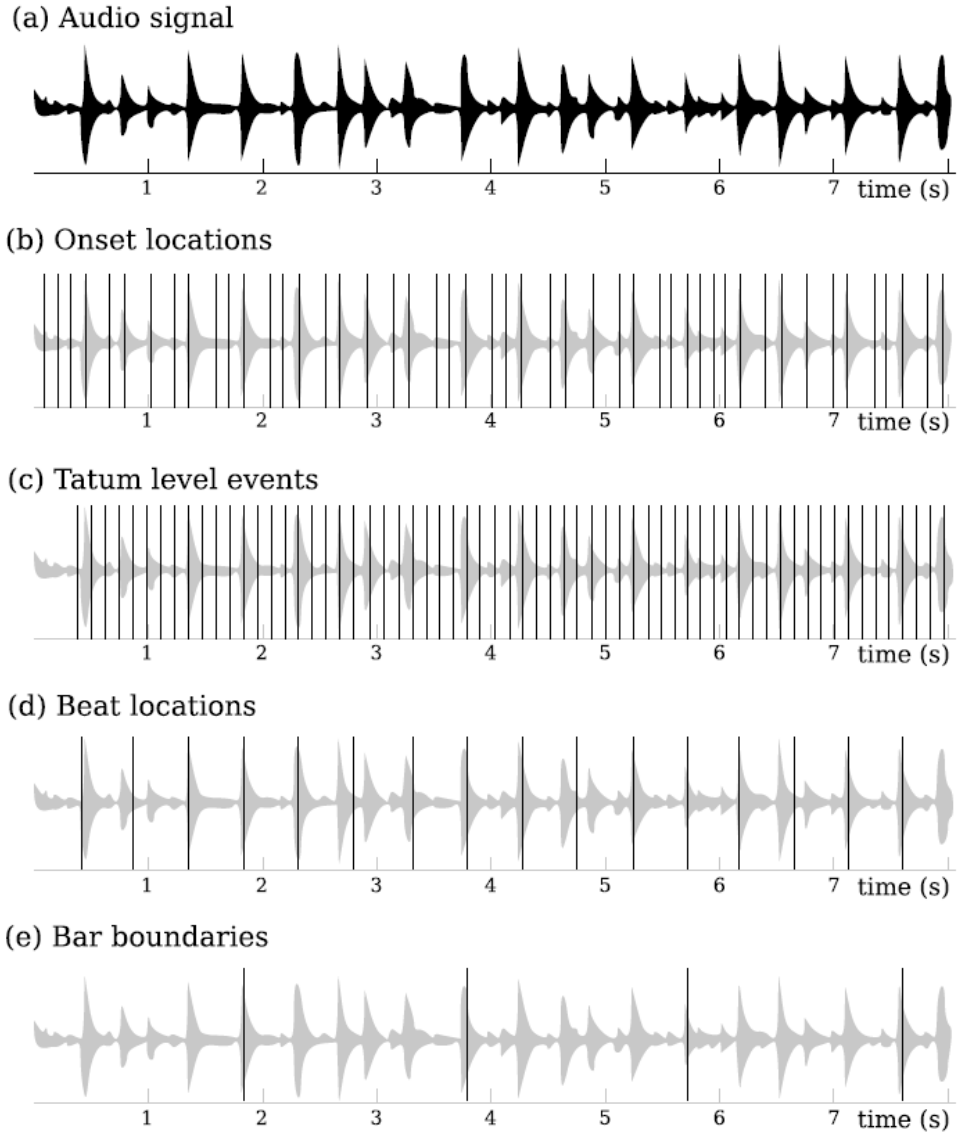


Figure 1.2: Metrical Structure for a Samba rhythm. (a) Audio signal. (b) Note onset locations. (c) Lowest metrical level: the Tatum. (d) Beat locations. (e) Bar boundaries. Example by Davies (2007).

Beat tracking can be considered one of the fundamental problems in music information retrieval (MIR) research. There have been numerous algorithms presented, *e.g.* Dixon (2007); Ellis (2007); Klapuri et al. (2006), whose common aim is to "tap along" with musical signals.

1.3. Applications of automatic rhythm description

There are several areas of research for which automatic rhythm description is relevant, like:

- Estimation of tempo and variations in tempo for performance analysis considers the interpretation of musical works, for example, the performer's choice of tempo and expressive timing. These parameters are important in conveying structural and emotional information to the listener (Clarke, 1999).

- Rhythm description is necessary for automatic score transcription from musical signals, like music transcription (Bello, 2003) , chord detection (Mauch et al., 2009), structural segmentation (Levy & Sandler, 2008).

- Rhythm data is used in audio content analysis for automatic indexing and content-based retrieval of audio data, such as in multimedia databases and libraries, like music similarity (Holzapfel & Stylianou, 2010), cover-song detection (Ravuri & Ellis, 2010).

- Automatic audio synchronization with devices such as lights, electronic musical instruments, recording equipment, computer animation and video with musical data. Such synchronization might be necessary for multimedia or interactive performances or studio post-production work. The increasingly large amounts of data processed in this way leads to a demand for automation, which requires that the software involved operate in a "musically intelligent" way, and the interpretation of beat is one of the most fundamental aspects of musical intelligence (Dixon, 2001).

Other applications

- Source Separation (Rafii & Pardo, 2013)

- Interactive music accompaniment (Robertson & Plumbley, 2007)

- Automatic remixing (Hockman et al., 2008)

- Real-Time Beat-synchronous Audio Effects (Stark et al., 2007)

- Biorhythms detection (Barabasa et al., 2012)

1.4. Overall scheme

The general scheme of automatic rhythm description methods proposed by Gouyon & Dixon (2005), presented in Figure 1.3, includes :

1. **Feature list creation block:** It transforms the audio waveform into a temporal series of features representing predominant rhythmic information.

2. **Pulse induction block:** It uses the parsed information to estimate periodicities in the signal.

   the following steps have also been incorporated by the beat tracking systems:

3. **Pulse tracking block:** It provides the temporal positions of the beats.

4. **Back-end block:** It parses the beat positions to a global tempo estimation or selects the strongest tempo for some methods. In order to compare all of the methods in the same conditions, this last block had to be implemented for some systems.

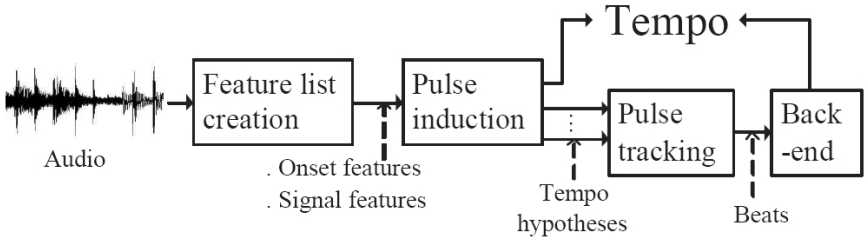


Figure 1.3: General tempo induction blocks by Gouyon & Dixon (2005).

1.5. Challenges

Automatic description of musical rhythm is not obvious. It seems to entails two processes: a bottom-up process, that enabling faster perception of pulses from scratch, and a top-down process (a persistent mental framework) that lets this induced perceptual guide the organization of incoming events (Desain & Honing, 1999). Implementing in a computer program both reactivity to the environment and persistence of internal representations is a challenge. It is important to say that rhythm description does not solely call for handling timing

features. Moreover, despite the somewhat automatic inclusion of beat trackers as temporal processing components in different applications, beat tracking itself is not considered a solved problem.
While the idea of a universal model for automatic rhythm description would seem to be an attractive goal, Collins (2006) proposes strong arguments as to why this is unrealistic. He suggests that the main flaw of computational beat tracking systems is a lack of understanding of the higher-level musical context; where this context is obvious to the trained human listener when tapping to music. The eventual route towards improving beat tracking would therefore appear to be through the use of higher level knowledge of musical style coupled with the understanding of how to apply this knowledge in the context of automatic rhythm description. For example, harmonic analysis (including tonality, key, and chord progressions) (Bello & Pickens, 2005; Gómez, 2006; Gómez & Bonada, 2005; Müller et al., 2005; Yoshii, 2008) can enhance the information of rhythm changes in music signals and improve the performance of automatic rhythm description algorithms (Eronen & Klapuri, 2010). Through simulated evaluation, e.g. in Davies & Plumbley (2005b), (Stark, 2011, ch.4), where *a priori* knowledge of the best beat tracking system per genre can be used, large hypothetical gains in performance are possible. However, to the best of our knowledge, no such system currently exists which can outperform the state of the art using automatic determination of musical style or genre, in order to select the best system per genre. Based on the above, improving automatic rhythm description (tempo and beat tracking) using multiple systems, entails two main challenges - determining which rhythm description systems to use and how this systems can be combined.

1.6. Goal and outline of the thesis

This thesis is driven by the hypothesis that when combined in a meaningful way multiple automatic rhythm description systems can complement each other to achieve better performance. In order to devise a method of combining automatic rhythm description systems, our principal goals to determine which rhythm description systems can be used and how can they be combined to improve the automatic rhythm description accuracies. We address an evaluation of tempo estimation and beat tracking approaches to analyze their capabilities and statistical relations, in order to choose a representative selection of these systems and propose a meaningful way to combine them and to build a system that can improve automatic rhythm description. Finally, based on the agreement of a committee of beat trackers, we proposed an automatic way to detect difficult samples for beat tracking and the use of voice suppression for improving beat tracking. Lastly we present a methodology for automatic beat tracking annotation of large data sets with a confidence value of the estimation. The work is divided into the following chapters.

Chapter 2

In this chapter we present an updated state of the art and comparative evaluation of automatic tempo estimation, in order to devise a method for tempo estimation using a combination of different approaches,. We consider 32 audio tempo estimation approaches, 28 academic and 4 commercial systems, in order to evaluate their respective accuracies and behavior on a subset of the music collection used in MIREX 2004 tempo task. We used this subset because allowed us to compare with approaches documented in the literature, where we did not have access to their software implementation. In the evaluation, we analyzed the differences between different steps of the algorithms. We provide their performance and error distribution, and discuss the strategies that seem to yield better results. We also proposed a tempo estimation system by selecting 7 tempo estimation approaches and using a decision tree method in order to improve the main performance of the best algorithm. Based on the evidence, the best tempo estimation performances in the evaluation are achieved by beat trackers.

Chapter 3

In order to devise a method for beat tracking using a combination of different approaches, we compiled and evaluated 16 state of art beat tracker systems. Based on their evaluation results we selected 5 beat tracker systems to build a committee, and we devised a meaningful way to compare the beat estimations of these beat tracking systems and we present an automatic method to select the best beat tracking estimation per song from a committee of beat trackers without the need of ground truth. This Method demonstrates a significant improvement over using individual state of the art beat tracking algorithms. Moreover, despite the good performance results of the beat tracking committee, it is a problematic approach because there are differences between approaches and system requirements of each algorithm. To sort this out, we extend the idea of the beat trackers committee in an implementable beat tracker system that uses the query by committee idea, using a committee composed by multiple onset detection functions as inputs to one beat tracker model, and the final output is selected from the beat estimations of the committee that more agree with the other ones. This proposed method outperforms the state of the art beat trackers in the evaluated measures.

Chapter 4

In chapter 3, we present a method for comparing the beat estimations of a committee of beat trackers by measuring the level of agreement between them. We found a correlation between the level of agreement and the mean accuracy performance of the beat trackers in the committee, using this information and based on the hypothesis that a glass ceiling in beat tracking exists due to a

lack of diversity in annotated data, we proposed a methodology to identify challenging music samples for beat tracking in a dataset without ground truth annotations. In order to improve beat tracking, we compiled a new public audio dataset for beat tracking evaluation that consists mainly of difficult pieces for beat tracking. We looked for the global audio properties that makes beat tracking difficult for the current of the state of the art systems with the intention of point out the difficulties and challenges for future work. One of the properties that makes beat tracking difficult is quiet accompaniment and strong vocals in songs, in order to improve the performance of existing systems, in this chapter we propose the use of voice suppression algorithms for music signals in the presence of highly predominant vocals. To evaluate this hypothesis, we compared systematically the accuracy and efficiency of five state of the art beat tracking systems against seven voice suppression systems. Finally, having demonstrated the validity of using the beat tracker output that most agrees among the committee of beat trackers to improve the mean performance in beat tracking on a manually annotated dataset, we now turn our attention to estimating the level of successful beat tracking without ground truth and applying it to a large collection of non-annotated data. We determine a threshold value in the agreement level of the committee to establish a value above which the beat tracker outputs are perceptually acceptable and the accuracy of the estimation is good.

Chapter 5

We conclude this thesis by summarizing the possible contributions to the field of each of the topics, and offer an analysis of the strengths and weaknesses of our proposed methods and suggest promises areas for further research.

1.7. Publications

This thesis contains work previously published in the following journals and conference papers:

ISI-indexed peer-reviewed journals

- Zapata, J.R., Davies, M.E.P., Gómez, E. **Multi feature beat tracking**. IEEE Trans. on Audio, Speech, and Language Processing.

- Holzapfel, A., Davies, M.E.P., Zapata,J.R., Oliveira, J.L., Gouyon, F. (2012). **Selective sampling for beat tracking evaluation**. IEEE Trans. on Audio, Speech, and Language Processing, 20 (9), 2539-2548.

Full-article contributions to peer-reviewed conferences

- Zapata, J.R. & Gómez, E. (2013). **Using Voice Suppresion Algorithms To Improve Beat Tracking In The Presence Of Highly Predominant Vocals**. In Proc. of the 38 IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP). Vancouver, Canada.

- Zapata, J.R., Holzapfel, A., Davies, M.E.P., Oliveira, J. L., F. Gouyon. (2012). **Assigning a confidence threshold on automatic beat annotation in large datasets**. In 13 Proc. of the Int. Conf. on Music Information Retrieval (ISMIR). pp. 157-162. Porto, Portugal.

- Zapata, J.R. & Gómez, E. (2012). **Improving Beat Tracking in the presence of highly predominant vocals using source separation techniques: Preliminary study**. In Proc. of the 9th Int. Symposium on Computer Music Modeling and Retrieval (CMMR). pp. 583-590. London, UK

- Zapata, J.R. & Gómez, E. (2011). **Comparative Evaluation and Combination of Audio Tempo Estimation Approaches**. In Proc. of the AES 42nd International Conference: Semantic Audio. pp. 198-207. Ilmenau, Germany.

- Holzapfel, A., Davies, M.E.P., Zapata, J.R., Oliveira, J.L., Gouyon, F. (2012). **On the automatic identification of difficult examples for beat tracking: towards building new evaluation datasets**. In Proc. of the 37 IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP). pp. 89-92. kioto, Japan.

- Bogdanov, D., Wack, N., Gómez, E., Gulati, S., Herrera, P., Mayor, O., Roma, G., Salamon, J., Zapata, J.R.& Serra, X. (2013). **ESSENTIA: an Audio Analysis Library for Music Information Retrieval**. 14th International Society for Music Information Retrieval Conference (ISMIR). pp. 493-498. Curitiba, Brazil.

- Bogdanov, D., Wack, N., Gómez, E., Gulati, S., Herrera, P., Mayor, O., Roma, G., Salamon, J., Zapata, J.R.& Serra, X. (2013). **ESSENTIA: an open-source library for sound and music analysis**. The 21st ACM International Conference on Multimedia. pp. 855-858 Barcelona, Spain.

Other contributions to conferences

- Zapata, J.R., Davies, M.E.P., Gómez, E. (2013). **MIREX 2013: Multi Feature Beat Tracker**. Music Information Retrieval Evaluation eXchange (MIREX) extended abstract.

- Zapata, J.R., Davies, M.E.P., Gómez, E. (2012). **MIREX 2012: Multi Feature Beat Tracker (ZDG1 AND ZDG2)**. Music Information Retrieval Evaluation eXchange (MIREX) extended abstract.
- Zapata, J.R. & Gómez, E. (2011). **Combination of Audio Tempo Estimation Approaches (MIREX 2011 Submission)**.Music Information Retrieval Evaluation eXchange (MIREX) extended abstract.

Beat Tracking Datasets

- **SMC Dataset:** Dataset with challenging beat tracking situations like: quiet accompaniment, expressive timing, changes in time signature, slow tempo, poor sound quality etc. By Holzapfel, A., Davies, M.E.P., Zapata, J.R., Oliveira, J.L., Gouyon, F.(2012). 217 Manually beat-annotated musical pieces.

  Genres:classical music, romantic music, jazz, blues, chanson, and solo guitar compositions.

  http://smc.inescporto.pt/research/data/

  This dataset was used in MIREX 2012 and MIREX 2013 Beat Tracking task.

  http://www.music-ir.org/mirex/wiki/2012:Audio_Beat_Tracking
  http://www.music-ir.org/mirex/wiki/2013:Audio_Beat_Tracking

- **DatasetVocal:** Dataset whose signal properties and highly predominant vocals of each excerpt makes beat tracking difficult for the state of the art systems. by Zapata, J.R. & Gómez, E. (2013).

  75 Manually beat-annotated musical pieces.

  Genres: romantic music, jazz, blues, chanson, swing, rock, folk, tango and Balkan music

  http://mtg.upf.edu/people/jzapata

Beat Tracker system

- **Multifeature Beat tracker**, open-source C++ audio beat tracker publicly available under Affero-GPL license, designed by Zapata, J.R., implemented in Essentia by Bogdanov, D.

  http://essentia.upf.edu/

  Algorithm: *BeatTrackerMultiFeature()*

1.8. Thesis contributions

The main contributions contained within this thesis are:

- Tempo Estimation:
  - Comparative evaluation of 32 state of the art tempo estimation algorithms.
  - A proposed tempo estimation algorithm that combines the tempo estimation of other algorithms and outperforms the results of the single evaluated approaches.
- Beat Tracking:
  - Evaluation of 16 state of the art beat tracking algorithms and the determination of the global properties that makes beat tracking difficult for the state of the art systems.
  - Automatic method for detect problematic audio songs for beat tracking without the need of ground truth.
  - Automatic method for selecting the best beat tracking estimation per song from a committee of beat trackers.
  - Public Audio Dataset with annotations (SMC Dataset) for beat tracking evaluation that consists mainly of difficult pieces for beat tracking.
  - Open source beat tracking system under GNU Affero public license, which uses the query by committee concept, that outperforms the beat tracking performance compared with 20 state of the systems with a confidence value of the estimation. Moreover, it can detect problematic audio songs for beat tracking without of ground truth.
  - Method for improving the beat tracking performance in music signals in the presence of highly predominant vocals using voice suppression systems.
  - Public Audio Dataset with annotations (DatasetVocal) for beat tracking evaluation that consist mainly of highly predominant vocal difficult pieces for beat tracking.

In order to guarantee the reproducibility of the results if this research, the scientific papers, built datasets, are available at:
`http://mtg.upf.edu/people/jzapata`

CHAPTER 2

Tempo Estimation

Many approaches to tempo estimation have been proposed in the literature, and some efforts have been devoted to their quantitative comparison. The first public evaluation of tempo extraction methods was carried out in 2004 by (Gouyon et al., 2006) evaluating the accuracy of 11 methods at the ISMIR audio description contest . In 2005, 2006, 2010, 2011, 2012 and 2013 the MIREX (Music Information Retrieval Evaluation eXchange) initiative[1] continued the evaluation of tempo extraction methods. In order to avoid the training of methods to the specific MIREX dataset, the audio files are not available to participants, so it is not possible to analyze limitations of the evaluated systems.

In order to devise a method for tempo estimation using a combination of different approaches , we present a state of the art and a comparative evaluation of automatic tempo estimation. We considered 32 audio tempo estimation approaches, 28 academic and 4 commercial systems, in order to evaluate their respective accuracies and behavior on a subset of the music collection used in ISMIR 2004 tempo task. We used this subset because it allow us to compare with approaches documented in the literature, where we did not have access to their software implementation.

In the tempo evaluation we analyze the differences between the different stages of the algorithms. We provide the systems performance and error distribution, and discuss the strategies that seem to yield better results. We also propose a combination method of some algorithms in order to improve the main performance of the best algorithm. Based on the evidence, a discussion of the limitations of current methods and ideas for future work are presented.

This chapter is structured as follows: first, we provide a brief description of the evaluated methods; second, the evaluation strategy, results, errors and statistical significance analysis are presented; third, a combination method of some tempo estimation algorithms is described and we end by providing our main conclusions on the limitations and challenges of the evaluated approaches.

[1] http://www.music-ir.org/

The Material of this chapter was previously published by Zapata & Gómez (2011)

2.1. Tempo estimation approaches

We have considered a total of 32 audio tempo induction methods, 11 of which were already evaluated in Gouyon et al. (2006). Five of them were already evaluated with the same dataset and the results with the same evaluation metrics are available in Eck & Casagrande (2005); Gainza & Coyle (2011); Gkiokas et al. (2010); Ong & Streich (2008); Peeters (2010), but each estimated tempo per song is not available, therefore, some results such as statistical significance are not accessible. We also tested 4 commercial stand-alone systems and we had access to 12 approaches through different infrastructures.
We provide a general description of each of the approaches. All of the approaches were used with default configuration parameters. Finally, in order to compare all of the outputs of the approaches under the same conditions, we implemented a stage that parses the outputs which are not only one value of BPM (Beats per minute), in the following way:

- Beat Positions are parsed into BPMs computing the IBI (inter beat interval), and the median value of all BPM's is taken as the system's output.
- MIREX tempo output: slower tempo (T1), a faster tempo (T2) and the strength of T1 relative to T2. The output is selected according to the value of the strength of T1 relative to T2 [0-1], so T1 is selected when the value is bigger than 0.5, T2 is selected in the other cases.

A summary of all of the algorithms is provided in Table 2.1

Aubiotempo

Aubio is an open source software released under the GNU/GPL license. The implementation for beat extraction based on Davies et al. (2005) is a test Vamp plugin for Sonic Annotator[2]. The feature extraction of this algorithm considers a complex domain onset detection function, and the pulse induction block computes the maximal output of passing the unbiased autocorrelation function (ACF), at the end uses a Context Dependent Model for beat alignment. The output of this system is the beat positions.

BeatIt

In 2006, Jordi Bonada and Fabien Gouyon from Music Technology Group, Universitat Pompeu Fabra, proposed an approach called BeatIt. It is a C++ implementation of a beat tracking algorithm. The input signal is split into several frequency bands. For each band, the energy is computed, compressed, and differentiated. Next, the peak-to-peak distances between the maximum peaks of the autocorrelation function of each band are computed and stored.

[2] www.omras2.org/SonicAnnotator

Those are added to a histogram of one BPM octave. The maximum of the histogram (the *tatum*, the fastest metrical level) sets the wrapped BPM estimation. Some statistics of the peak-to-peak distribution are used to select the output BPM octave. The output is a single tempo value.

Beatroot

BeatRoot, developed by Dixon (2001) when he was working at OFAI Intelligent Music Processing Group, is a java implementation for beat tracking under the GNU Public License [3]. This algorithm is based on a spectral flux onset detector followed by an inter onset interval (IOI) clustering algorithm. The output of this algorithm is the beat positions. We also consider another 3 algorithms from the same author (DixonI, DixonT, DixonACF). These algorithms were evaluated in the audio tempo induction task at ISMIR 2004 by Gouyon et al. (2006).

BpmHistogram

Aylon & Wack (2010) from the Music Technology Group (UPF), proposed a tempo estimation approach based on the Predominant Local Pulse curves (PLP), by Grosche & Müller (2009). The method assumes a constant tempo in the song, in order to be able to better estimate the beat locations where the confidence is low. The algorithms computes the PLP curve (combination of weighted novelty curves derived from the first order difference energy curves of 5 bands), and the autocorrelation function is used in the pulse induction process, then a histogram of the principal peaks over tempo are calculated and finally the prominent peak in the pulse induction block is used as output of the algorithm. This method is available at the Essentia framework[4]

Eck

Eck & Casagrande (2005) from The University of Montreal, proposed a tempo induction algorithm based on the detection of the metrical structure. First the audio signal is down-sampled, then the sum-of-squares of the envelope is computed over windows of size 42 with 5 points of overlap. The periodicity function detection is then calculated by autocorrelation plus entropy on the phase autocorrelation matrix, and multiple hierarchically related lags in prediction. The tempo selection is done by an analysis over the hierarchical meter relations between peaks.

[3] www.eecs.qmul.ac.uk/~simond/beatroot/

[4] http://essentia.upf.edu/ BpmHistogram()

Ellis

Ellis (2007) from Columbia University, proposed a tempo induction approach on Matlab [5]. The algorithm computes an onset energy envelop obtained from a 40 mel-frequency spectrogram as audio feature, and for the Pulse induction block an autocorrelation function is computed over the onset envelop to obtain the periodicity peaks. The output of this algorithm is given as the MIREX audio tempo estimation task (slower tempo and faster tempo).

Fixedtempo

Cannam from the Centre for Digital Music Queen Mary, University of London, wrote this "Simple Fixed Tempo Estimator" as a simplification of the method derived from work by Davies & Plumbley (2005a). This algorithm is part of the vamp examples plugins in Sonic annotator[6]. This algorithm analyzes a fragment of audio and estimates its tempo. Assuming an input of fixed tempo, it analyzes only the first seconds before returning a result, discarding all subsequent input. The audio feature block calculates an overall energy rise function across a series of short frequency-domain input frames, and then in the pulse induction part takes the autocorrelation of this function, filters it to stress possible metrical patterns, locates peaks, and converts them from autocorrelation lag to the corresponding tempo from the pulse induction block. A simple perceptual curve is also applied in order to increase the probability of detecting a "likely" tempo in the filtering step. For improved tempo precision, each tempo with strong related peaks is averaged for the tempo calculated from those peaks. The output of the algorithm is a single tempo value.

GK

Gkiokas et al. (2010), from the National Technical University of Athens, proposed a tempo induction approach using filter-bank analysis and tonal features and assuming a constant tempo. For the audio feature block a sliding window is applied to the signal and two feature classes are extracted, the log-energy of each band of a mel-scale triangular filter-bank, and the strengths of the twelve western musical tones at all octaves for each audio frame. The pulse-induction was carried out by convolution of the time-evolving feature vectors with a bank of resonators, each resonator corresponding to a target tempo. Then the results of each feature class are combined to give the final output.

Hyb2

Gainza & Coyle (2011) from the Dublin Institute of Technology, proposed an algorithm called *Hyb2*. The algorithm splits the signal into three different

[5] labrosa.ee.columbia.edu/projects/beattrack/tempo2.m

[6] www.vamp-plugins.org/plugin-doc/vamp-example-plugins.html#fixedtempo

frequency bands; then detects the signal changes using a complex spectral onset detection method, calculated from the low frequency bands. The transient detection method is calculated from the mid and high frequency bands. The periodicity detection is based on autocorrelation in each band and it is then combined into a single periodicity function. A weight function is applied in order to reduce the number of double and half tempo estimations.

IBT

Oliveira et al. (2010) proposed a C++ approach for tempo induction and beat tracking system based on the strategy of competing agents sequentially processing musical input, and considering parallel hypotheses regarding tempo and beats, this strategy was introduced by Dixon (2001) in the BeatRoot system. It differs from BeatRoot strategy by using a causal decision process over competing agents, instead of taking decisions after the whole data that has been analyzed. Spectral flux is used for the audio feature extraction, then the algorithm implements a period hypotheses induction and phase hypotheses selection, this is followed by an agents Setup to score each hypothesis and to rank them. IBT is integrated in MARSYAS 0.4.0 framework `marsyas.info/`, under GPL general public license. The algorithm was tested in offline mode and gives a BPM estimate value as one of its outputs.

jAudio

McKay from McGill University, Canada, is the author of the tempo induction algorithm called StrongestBeat of BeatHistogram[7] and implemented in the system Jaudio 1.0.4. using the Java framework: `jaudio.sourceforge.net/`. The algorithm extracts the energy envelope to obtain the Beat Histogram from a signal. This histogram shows the strength of different rhythmic periodicity in a signal and is calculated by taking the RMS of 256 windows and then taking the FFT of the result, at the end the BPM is calculated by finding the strongest beat and dividing it by the sum of all entries in the beat histogram. The output of the tempo estimation is a single tempo value.

MIRTempo

Lartillot (2010) from University of Jyväskylä, provides the Mirtempo algorithm as part of the Mirtoolbox platform [8]. The feature extraction consists of an onset curve, represented by an amplitude envelope computed through a 10-channel Gamma-tone filter bank and a low-pass filtering. This procedure retains from the signal the long-term evolution, while filtering faster oscillations.

[7] `jaudio.sourceforge.net/jaudio10/javadoc/jAudioFeatureExtractor/AudioFeatures/BeatHistogram.html`

[8] `www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/mirtoolbox`

The signal is down-sampled and subsequently differentiated by computing the difference between successive samples. The pulse induction process is based on autocorrelation in a tempo range between 20 and 230 BPM. Tempo strength is estimated using the normalized autocorrelation coefficients related to the estimated periodicity, and the output is the one with the highest tempo strength. The MIR toolbox is available under the GNU general public license.

MPEG7-xm

Jan Rohden from the Fraunhofer Institute for Digital Media Technology IDMT, described this matlab algorithm[9] along MPEG document w5212 (15938-4:2001 /FPDAM). The algorithm starts by extracting energy using the same process proposed by Scheirer (1998). It considers 6 frequency bands in segments of 4 sec. each. The envelopes obtained for each band are then weighted and the pulse induction is based on autocorrelation based periodicity detection via forward and inverse fft (biased autocorrelation). The output is a BPM value each time the estimated tempo changes. For our evaluation, the estimated BPM of each song is computed as the median of all the provided estimations.

OS

Ong & Streich (2008) from the Center for Advanced Sound Technologies, Yamaha, created an approach for tempo induction called OS [an efficient offline beat tracking method for music with steady tempo]. The audio feature is based on an enhanced onset detection function from the spectral differences between adjacent frames. The pulse-induction block is based on autocorrelation, and of the different candidates, the output is the most reliable peak. This tempo value is also used to initialize the beat tracking algorithm.

Peeters

Peeters (2010), from IRCAM, proposed an algorithm which uses reassigned spectral energy flux with a window of 93ms in the audio feature block. The tempo-induction step is based on a new periodicity measure consisting of a combination of discrete Fourier transform and frequency-mapped autocorrelation function. Using a set of proposed meter/beat subdivision templates a Viterbi decoding algorithm estimates the most likely tempo and meter over time.

QMTempo

Davies and Landone from the Centre for Digital Music Queen Mary, University of London, proposed an algorithm for tempo and beat tracking called qmtempotracker based on Davies & Plumbley (2007) beat tracker, which works as

[9] mpeg7.doc.gold.ac.uk/mirror/v2/Matlab-XM/AudioBpmD/AudioBpmD.m

a Vamp plugin in SonicAnnotator [10]. We used the "Complex domain" method for our tests, which consists of a hybrid of the two-state beat tracking model and a dynamic programming method. It computes the onset detection function to estimate the tempo contour and then given the tempo, to recover the beat locations.
The periodicity estimation is based on autocorrelation, this signal is weighted by a filter-bank and grouped together into a matrix of observations of periodicity through time. The best path of periodicity is found using the Viterbi algorithm, where the transition matrix is defined as a diagonal Gaussian. Given the estimates of periodicity, the beat locations are recovered by applying a dynamic programming algorithm. Its output consists of BPM values for each time the estimated tempo changes. For our test, estimated BPMs are combined into a single value using the median.

Tzanetakis

Tzanetakis (2010) from The University of Victoria proposed a tempo induction algorithm for MARSYAS[11] framework distributed under the GNU Public Licence (GPL) and presented in MIREX 2010. The audio tempo estimation is based on transforming a normalized autocorrelation of the onset strength signal (based on spectral flux) to a Beat Histogram. A simple peak picking heuristic is used to select the dominant tempo. This algorithm provides a MIREX output. Tzanetakis is also the author of 3 other algorithms (TzanetakisMS, TzanetakisMM and TzanetakisH) that took part at the previously experimental comparison of audio tempo induction algorithms at ISMIR 2004.

Algorithms of ISMIR Audio Description Contest 2004

The evaluation presented here has been carried out using the audio excerpts dataset from the ISMIR Audio Description Contest 2004[12] by Gouyon et al. (2006). This allows us to compare the results of new methods with the ones that were evaluated before:

- AlonsoACF, AlonsoSP (Alonso et al., 2004)
- DixonACF (Dixon & Pampalk, 2003), DixonI and DixonT (Dixon, 2001)
- Klapuri (Klapuri et al., 2006)
- Scheirer (Scheirer, 1998)
- TzanH, TzanMS and TzanMM (Tzanetakis & Cook, 2002)
- Uhle (Uhle et al., 2004)

[10] www.vamp-plugins.org/plugin-doc/qm-vamp-plugins.html#qm-tempotracker
[11] marsyas.info/
[12] mtg.upf.edu/ismir2004/contest/tempoContest/data3.tar.gz

Commercial tempo systems

We selected four commercial systems to compare the performance of academic approaches with theses, but the information about how these methods work is not documented.

Auftakt

AufTAKT V2 is a tempo estimator of music audio signals where audio signals are analyzed in terms of onset information (note on) via an algorithm which detects energy and frequency components and weighting them according to their perceptual importance. This system is able to adapt the tempo and beat estimation to an input signal with varying tempo.
www.zplane.de/index.php?page=description-Auftakt.

Beatcounter

BPM Counter, from Abyssmedia, is a free stand-alone beats per minute detector for MP3 music for windows www.abyssmedia.com/bpmcounter/.

Beatunes

Beatunes from Tagtrum industries, is a bpm estimator for itunes, which can be used as a 14-day trial or with a commercial license, www.beatunes.com/.

BPMer

BPMer from wildbits is a bpm estimator for Macintosh OS and its binary can be evaluated for free. www.wildbits.com/bpmer/.

2.2. Evaluation

2.2.1. Dataset

The *Song excerpt* dataset is a part of the datasets used for the ISMIR 2004 tempo induction[13] contest presented in Gouyon et al. (2006). It consists of songs with approximately constant tempi, and the format is the same for all: mono, linear PCM, 44100 Hz sampling frequency, 16 bits resolution. The total duration of the test set is approximately 9300 sec. The dataset is composed of 465 song excerpts of 20 seconds. The genre distribution is in Table 2.2 and a tempo range between 24 and 242 bpm, Figure 2.1. The ground-truth tempo was computed as the median of the IBIs (Inter Beat Interval).

[13] http://mtg.upf.edu/ismir2004/contest/tempoContest/data3.tar.gz

Algorithm	AlonsoACF	AlonsoSP	Aubio	BeatIt	Beatroot	DixonACF
Author	Alonso et al. (2004)	Alonso et al. (2004)	Brossier Davies et al. (2005)	Bonada and Gouyon	Dixon (2001)	Dixon & Pampalk (2003)
Software	Matlab	Matlab	Vamp	Windows	Java	Matlab
Output	One Bpm	One Bpm	Beats in time	One Bpm	Beats in Sec.	One Bpm
Audio Feature	Onsets of Notes	Onsets of Notes	Complex spectral difference	Energy differences in 8 bands	Energy based Onset detector	Energy of 8 freq. bands
Periodicity	ACF	SP	ACF	ACF	IOI	ACF

Algorithm	DixonI	DixonT	Eck	Ellis	BpmHist	Fixedtempo
Author	Dixon (2001)	Dixon (2001)	Eck & Casagrande (2005)	Ellis (2007)	Aylon & Wack (2010)	CannanDavies & Plumbley (2005a)
Software	Java	Java		Matlab	Essentia	Vamp
Output	One Bpm	One Bpm		slower and faster tempo	One Bpm	One Bpm
Audio Feature	Energy based Onset Detector	Energy based Onset Detector	The sum of squares of the envelope	Mel Auditory Feature	Energy enveloped differences for 5 bands	Overall energy rise function.
Periodicity	IOI	IOI	ACF	ACF	ACF	ACF

Algorithm	GK	Hyb2	IBT	jAudio	Klapuri	MIRTempo
Author	Gkiokas et al. (2010)	Gainza & Coyle (2011)	Oliveira et al. (2010)	McEnnis and McKay	Klapuri et al. (2006)	Lartillot (2010)
Software	Matlab		Marsyas	Java	Linux	Matlab
Output	One Bpm	One Bpm	One Bpm	One Bpm	One Bpm	One Bpm
Audio Feature	Spectral Flux	Spectral Flux	Spectral Flux	Energy envelope (256 window)	Loudness difference in 36 freq. subbands	10 channel gammatone filterbank
Periodicity	ACF	ACF	ACF	ACF	BF	ACF

Algorithm	Mpeg7-xm	OS	Peeters	Qmtempo	Scheirer	Tzanetakis
Author	Rohden	Ong & Streich (2008)	Peeters (2010)	Davies & Plumbley (2007)	Scheirer (1998)	Tzanetakis (2010)
Software	Matlab	Matlab	C/C++	Vamp	Linux	Marsyas
Output	BPM values when tempo changes			BPM values when tempo changes	Beats in sec.	slower and faster tempo
Audio Feature	Energy from 6 bands	spectral differences	Reasigned spectral flux	Spectral Flux	Energy differences for 6 bands	Onset strength signal
Periodicity	ACF	ACF	DFT /FM-ACF	ACF	BF	ACF

Algorithm	TzanH	TzanMM	TzanMS	Uhle
Author	Tzanetakis & Cook (2002)	Tzanetakis & Cook (2002)	Tzanetakis & Cook (2002)	Uhle et al. (2004)
Software	Linux	Linux	Linux	Windows
Output	One Bpm	One Bpm	One Bpm	One Bpm
Audio Feature	Energy in 5 octave space freq. bands using Wavelets	Energy in 5 octave space freq. bands using Wavelets	Energy in 5 octave space freq. bands using Wavelets	Energy difference from log. space freq. bands
Periodicity	ACF	ACF	ACF	ACF

Table 2.1: Brief description of all methods (ACF= Autocorrelation, BF= Bank-comb filter, SP= Spectral product, IOI = inter onset interval clustering, DFT/FM_ACF = Discrete Fourier transform and frequency_mapped ACF)

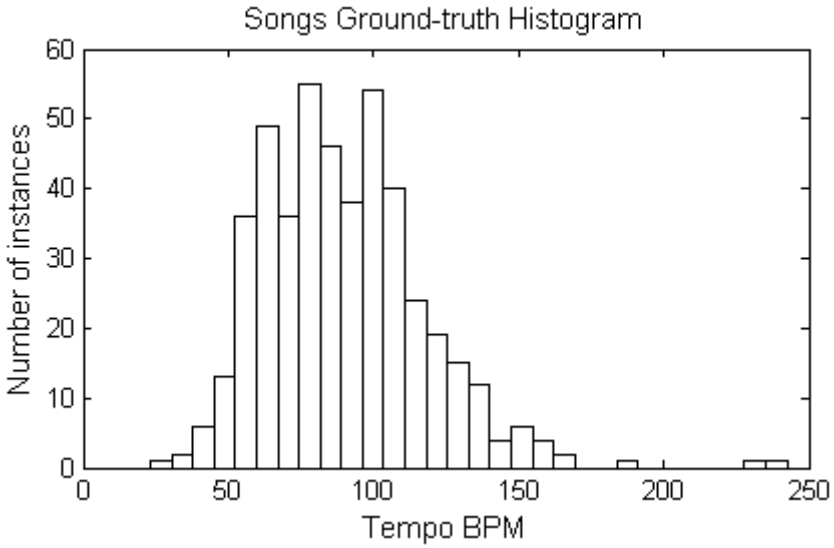


Figure 2.1: BPM ground-truth Histogram.

Table 2.2: Genre Distribution of the song excerpts

Genre	# Songs
Rock	68
Classical	70
Electronic	59
Latin	44
Samba	42
Jazz	12
Afrobeat	3
Flamenco	13
Balkan and Greek	144
Fado	10

2.2.2. Tempo measures

This evaluation was carried out using a single tempo estimation value for each algorithm, this allowed us to compare the previous evaluation in ISMIR 2004 to the existing results in published research. The evaluation metrics were then used for the test:

- **Metric 1:** The percentage of the tempo estimation within a 4% (precision window) of the ground truth. This procedure was used to evaluate the accuracy of the algorithms to detect the main bpm of the song.

- **Metric 2:** The percentage of the tempo estimation within a 4% (precision window) of the 1, 2, $\frac{1}{2}$, 3, $\frac{1}{3}$ times the ground-truth. This procedure takes into account the problems of double or triple deviation of the tempo estimation.

2.2.3. Results

The mean performance results of the evaluation metrics 1 and 2 are summarized in Table 2.3 and Figure 2.2 including the combination method explained in section 2.3. Table 2.4 shows the overall ranking of algorithms according to the estimation accuracy (metric 1 and 2). We highlighted some characteristics of the algorithms that we consider relevant: the strategy for pulse induction and, for those methods using multi-band processing, it's specified if the band integration happens before or after the periodicity detection stage.
By using the oracle result (the best estimation per song from the results of all the methods), it is possible to reach an accuracy of: [90.53%, 100%] in the evaluation metrics 1 and 2 respectively. As a general observation, for all the song excerpts, at least one algorithm correctly estimates the tempo with a ratio of 2, $\frac{1}{2}$ or 3. For these reasons, we conclude that rhythmic periodicity can be accurately estimated from the raw audio signal. Also the maximum number of algorithms agreeing on the same correct tempo estimation is 24, and this occurs for 3 song-excerpts. This means that none of the songs was extremely easy for all algorithms to estimate the tempo correctly at the same time.

Table 2.3: Tempo evaluation results

	AlonsoACF	**AlonsoSP**	**Auftak**	**Aubio**	**Beatcounter**
Metric 1	23.44	37.42	56.13	39.35	32.90
Metric 2	58.28	68.60	83.44	67.31	49.68
	BeatIt	**Beatroot**	**Beatunes**	**BPMer**	**BpmHistogram**
Metric 1	**60.43**	23.23	19.35	31.83	24.52
Metric 2	78.28	67.96	38.28	63.01	83.44
	DixonACF	**DixonI**	**DixonT**	**Eck**	**Ellis**
Metric 1	16.99	28.60	19.35	**60.00**	45.59
Metric 2	76.99	62.58	68.82	79.00	80.65
	FixedTempo	**GK**	**Hyb2**	**IBT**	**jAudio**
Metric 1	24.73	42.15	48.90	35.91	5.16
Metric 2	50.75	**90.11**	**91.80**	79.78	32.26
	Klapuri	**MIRTempo**	**Mpeg7-xm**	**OS**	**Peeters**
Metric 1	**58.49**	30.97	48.39	42.20	49.50
Metric 2	**91.18**	65.59	70.54	70.50	83.70
	Qmtempo	**Scheirer**	**Tzanetakis**	**TzanH**	**TzanMM**
Metric 1	43.23	37.85	25.59	21.29	18.71
Metric 2	80.43	69.46	66.45	47.74	41.08
	TzanMS	**Uhle**	**Combination**		
Metric 1	27.53	41.94	**65.37**		
Metric 2	52.47	71.83	**91.39**		

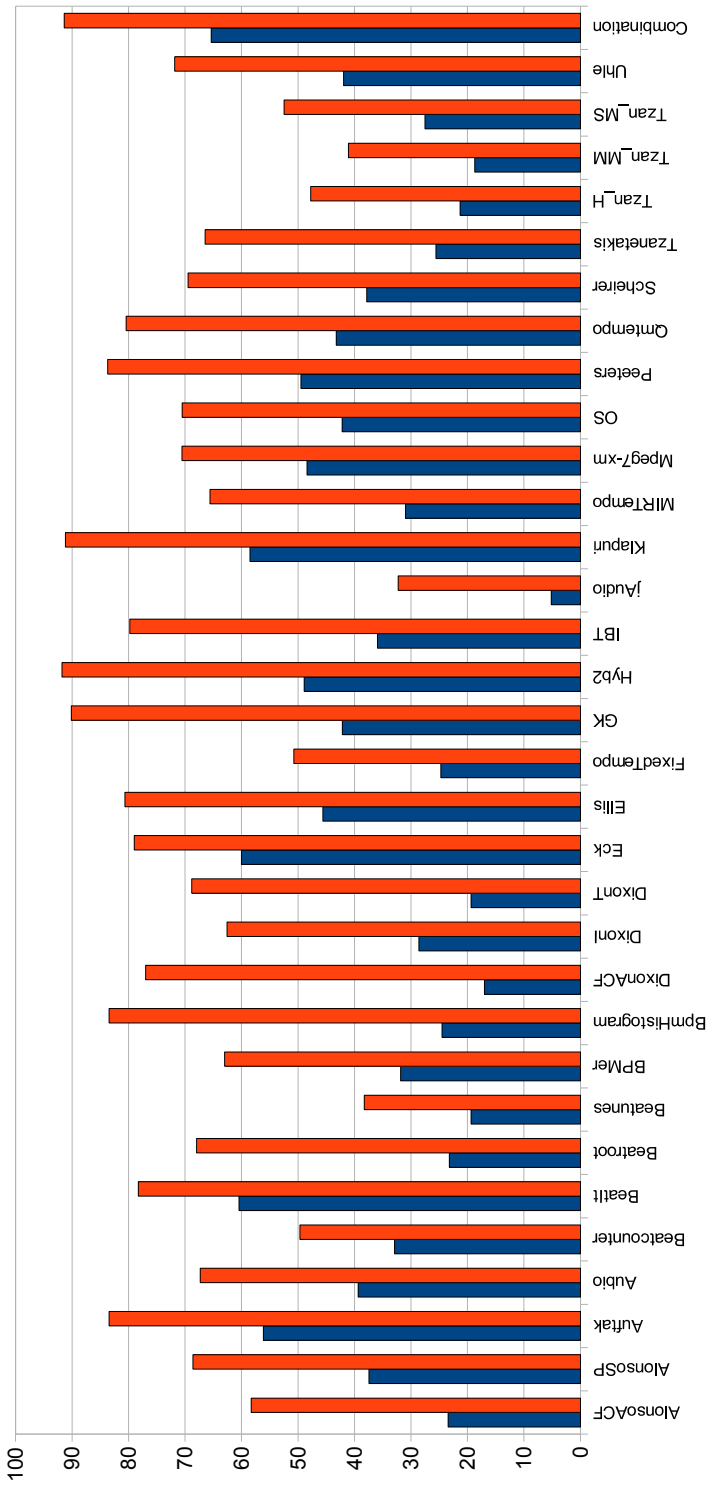


Figure 2.2: Tempo Evaluation Results, Blue bar Metric 1 and Orange bar Metric 2.

Table 2.4: Evaluation performance ranking of methods and Periodicity data (ACF = Autocorrelation, BF = Bank-comb filter, SP= Spectral product, IOI = inter onset interval clustering, DFT/FM_ACF = Discrete Fourier transform and frequency_mapped ACF), Band Combination (after or before pulse induction) and Beat estimation

ALGORITHM	Pulse Induction	Combining Bands	Beats
Klapuri	BF	After P.I	Yes
Hyb2	ACF	After P.I	
Eck	ACF		Yes
BeatIt	ACF	After P.I	Yes
Peeters	DFT/FM- ACF		Yes
GK	BF		
Ellis	ACF		
Qmtempo	ACF		
Mpeg7-xm	ACF	After P.I	
IBT	ACF		Yes
Uhle	ACF	After P.I	
OS	ACF		Yes
BpmHistogram	ACF	After P.I	Yes
Scheirer	BF	After P.I	Yes
Aubio	ACF		Yes
AlonsoSP	SP		
MIRTempo	ACF	Before P.I	
DixonACF	ACF	Before P.I	
Tzanetakis	ACF		
Beatroot	IOI		Yes
DixonI	IOI		
DixonT	IOI		
AlonsoACF	ACF		
TzanMS	ACF	Before P.I	
FixedTempo	ACF		
TzanH	ACF	Before P.I	
TzanMM	ACF	After P.I	
jAudio	ACF		

The best performance in the evaluation is obtained by the Klapuri algorithm. It obtains the following accuracy measures: [58.49%, 91.18%]. The first metric is lower than for BeatIt and Eck [60.43%, 60%] respectively, but the results are statistically comparable for BeatIt and Klapuri. For metric 2, Hyb2 provides the best performance 91.8% but with a small difference 0.72% with Klapuri, so that the statistical significance is not considerable.
The main difference between Klapuri's method and the others lies in the audio signal feature, which computes the subtle energy changes that might occur in narrow and wide frequency sub-bands, but the induction block is a bank-comb filter like the one used by Scheirer (1998). So we might assume that the accuracy of the algorithm lies in good feature extraction, rather than in a complex tempo induction block. At the end, Klapuri's algorithm computes the tempo as the median of the IBI's. Being the same method used to compute the ground truth of the music collection.
From the information of each approach and by comparing 12 algorithms (Klapuri, Hyb2, BeatIt, Mpeg7-xm, uhle, BpmHistogram, scheirer, DixonACF, MIRtempo, Tzan_ms, Tzan_h, Tzan_mm), we observed that these algorithms divides the audio signal into sub-bands and then uses autocorrelation in the pulse induction block. The seven best performing algorithms compute the autocorrelation function before the frequency integration step. The statistical difference of the first three algorithms and the other methods, which compute autocorrelation after combining signals of each band, are significant.
The algorithms which compute the first derivative of the signal (Klapuri, BeatIt, Uhle, BpmHistogram and Scheirer) in the audio feature performed better than those that employ only frame values.
None of the algorithms correctly estimated the tempo of the 13 song-excerpts with a tempo below of 49 bpm, but more than 4 methods provided the double or triple of the ground-truth annotation. This might reflect that current methods are not adapted to slow tempi, but they can detect at least one metrical level of the song, mostly the *tatum*. For the 36 song-excerpts with only one correct estimation (ratio=1), the algorithms with most correct estimations were BeatIt (seven) and IBT (five), and the double tempo was the most common error among the rest of algorithms.
We also observe significant differences in the values of metric 1 and metric 2, because most of the algorithms detect the double or the triple tempo of the ground-truth. At least 22 algorithms are above 65% accuracy for metric 2.

2.2.4. Statistical significance

The statistical significance of the algorithm estimations in the two evaluation metrics was carried out by means ofthe McNemars Test[14] (Gillick & Cox, 1989), considering a p-value of 0.01 as the threshold for statistical significance.

[14] staffwww.dcs.shef.ac.uk/people/R.Hofe/mcnemar.html

The significance between algorithms for each evaluation order by the performance in the whole evaluation is presented on Figure 2.3. Metric 1 and 2 are respectively in the bottom and top side of the main diagonal. From the statistical comparison between algorithms, a filled cell represents an equal statistical performance between the algorithms.
According to this statistical significance analysis, Beatroot, Tzanetakis and DixonI results were found to be comparable to each other. Moreover, Aubio, AlonsoSP, Scheirer and Uhle had the same statistical performance too. The relation between DixonI and Beatroot shows that the difference in the pulse induction block does not represent statistical differences in the results. The best performance in metric 1, obtained by BeatIt method and is statistically comparable with the Klapuri method; however it shows lower performance in metric 2. The difference between these algorithms results, is a consequence of the lower tempo estimation tendency in the BeatIt algorithm, because its octave correction processing.

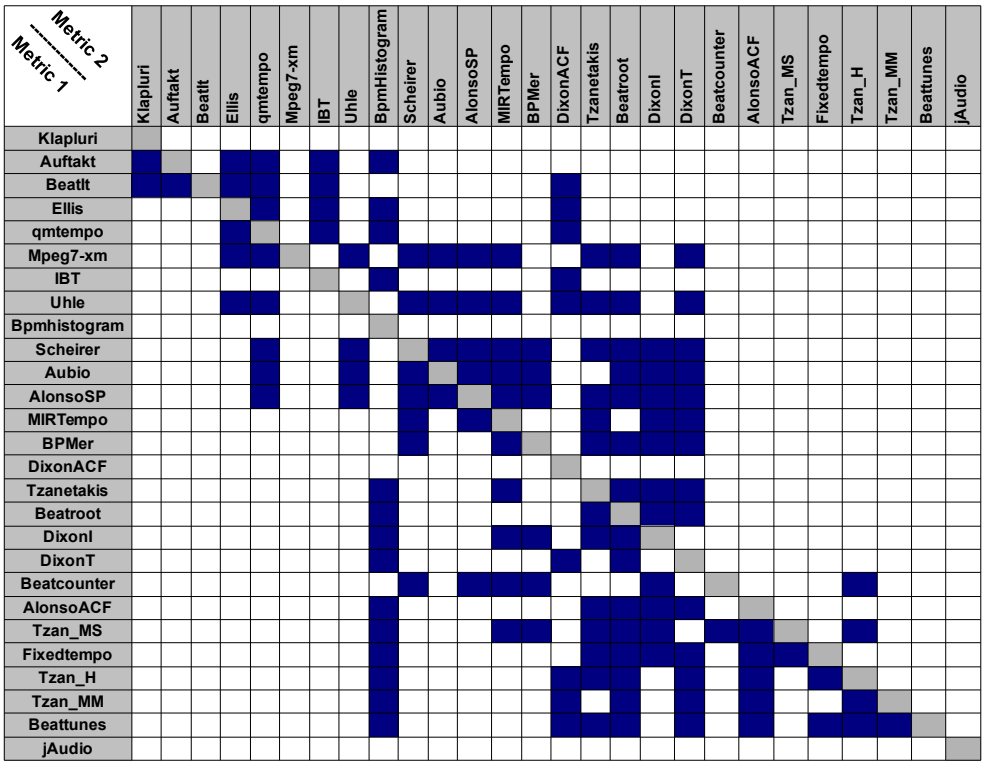


Figure 2.3: Statistical significance between tempo algorithms in Metric 1(low side) and Metric 2 (up side), a filled cell represents an equal statistical performance between the algorithms.

2.2.5. Error analysis

Compared to the ground truth of the dataset, most of the algorithms estimates the Double, but other error tendencies such as $\frac{1}{2}$, 3, 4, $\frac{4}{3}$, $\frac{2}{3}$ were present in the results of all of the algorithms in the whole dataset. The test was done without any knowledge of the meter of the songs, but a ternary tendency of some songs can be detected from the relation between the algorithm estimations. The histogram of the error ratio tendencies and the values of all of the algorithms can be seen in Figure 2.4 and Table 2.5.
We first observe that seven algorithms (Beatunes, FixedTempo, Tzanetakis, MIRtempo, Aubio, AlonsoACF, Tzan_mm) had an error with a ratio = $\frac{4}{3}$ above 8% with a value of [9.68%, 12.9%, 11.40%, 10.75%, 10.54%, 8.6%, 8.39%] respectively. This represents an error of $\frac{3}{4}$ in the Inter Beat Intervals, that is, a focus on e.g the dotted quarter-note instead of the half- note. This error is more common than the $\frac{1}{2}$ in this dataset.

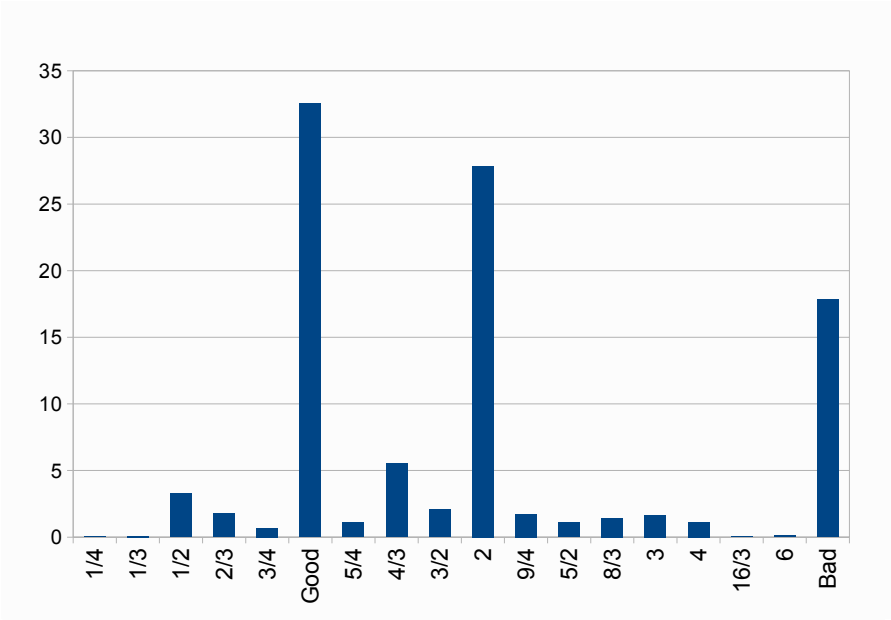


Figure 2.4: General error ratio histogram

The Jaudio algorithm tends to estimate faster tempo than the rest, and has most of its estimations above a ratio of 2. Because this algorithm was tested with default parameters, we cannot conclude if it tends to estimate faster tempi than the others. On the other hand, the BeatIt algorithm tends to estimate slower tempo than the rest, but the tempo distribution of the dataset had more song-excerpts with a BPM in a range between 60 and 110, so a dataset with equal bpm distribution would be needed to confirm this tendency.

The sum in percentage of the fraction errors of all of the algorithms is 18.40%. If the meter of each song excerpt was known,the tempo estimation could then be improved. The errors of $\frac{2}{3}$, $\frac{4}{3}$, $\frac{8}{3}$ and 3 show a problem in the pulse selection. Without this knowledge it can be difficult to estimate the ground- truth value. These estimations can be useful for meter detection and are a problem in the pulse selection process, but they show that the methods can detect a periodicity related with the ground-truth in the audio signal.
Based on the differences of evaluation metric 1 and 2, the algorithms with tendency to detect the double or the triple value instead of the ground- truth values are: DixonAFC, BpmHistogram, DixonT, jAudio, Beatroot, Tzanetakis, AlonsoACF, Tzan_MM and MIRtempo, with a difference equal to [42.58%, 32.9%, 29.89%, 21.94%, 21.08%, 12.04%, 8.6%, 1.08%, 0.86%] respectively.

Table 2.5: Other tendencies ratio results of all of the algorithms (Figure 2.4)

Ratio	**%**
1/4	0.02
1/3	0.09
1/2	3.29
2/3	1.77
3/4	0.68
Good	32.55
5/4	1.12
4/3	5.50
3/2	2.06
2	27.85
9/4	1.73
5/2	1.09
8/3	1.43
3	1.66
4	1.15
16/3	0.06
6	0.13
Other	17.81

2.3. Combination of methods for tempo estimation

Based on the statistical significance, we used the estimation results of the top seven academic methods in the evaluation (Klapuri, BeatIt, Ellis, Qmtempo, IBT, BpmHistogram and Mpeg7-xm) as components in a heuristic decision tree strategy to improve the results of the best method in the evaluation.

The hypothesis obtained from the experiment is that the ground-truth and the double tempo value are among the estimations for most of the song-excerpts. The Klapuri estimations are checked with a metrical hierarchy analysis using the results of the other six methods, and using the low tempo tendency of BeatIt and the double tendency of BpmHistogram, In order to correct inaccurate estimations and meter level errors. The steps are presented in pseudo code in the algorithm 1. The accuracy results of this configuration are: [**65.37%**, **91.39%**] for metric 1, 2 respectively in this dataset. These values are less than the oracle results but Klapuri estimations are improved in 6.67% for metric 1 and 0.21% for metric 2. Calculating the statistical relation between Klapuri and the combination method the p-values are: [**0.0029**, **1**] for the evaluation metrics 1 and 2 respectively, considering a p-value of 0.01 as threshold for statistical significance, the results of the combination method against Klapuri are statistical different for metric 1.

Algorithm 1 BPM = Combination-BPM(file.wav)

% BPM Estimations
BeatIt = BeatIt-BPMestimation(file.wav);
Ellis = Ellis-BPMestimation(file.wav);
BpmHistogram = BpmHistogram-BPMestimation(file.wav);
Klapuri = Klapuri-BPMestimation(file.wav);
Qmtempo = Qmtempo-BPMestimation(file.wav);
IBT = IBT-BPMestimation(file.wav);
Mpeg7-xm = Mpeg7-xm-BPMestimation(file.wav);
array = [BeatIt Ellis Qmtempo IBT BpmHistogram Mpeg7-xm];

% Combination and Selection
if (Klapuri ≠ BeatIt && Klapuri ≠ BpmHistogram) **then**
  if three or more values from **array** are equal **&** at least one value in the **array** are the double of the repeated value **then**
    Out = the repeated value in the array;
  else
    Result = Klapuri;
  end ifthat
else
  Result = Klapuri;
end if

% To avoid the double error
if (Result == BeatIt*2 **&** BeatIt == Ellis) **then**
  Result = BeatIt;
end if
return Result

Some of these tested methods had more than one output and the combination results are less than the oracle results, if all outputs of the methods are used, assuming these results as some other tempo estimator, the tempo estimation might be improved.

2.3.1. Tempo estimation submission (MIREX 2011)

Based on the oracle results, the *tatum* and *tactus* tempo hierarchal levels can be obtained from the tempo estimations of all these approaches for the evaluated dataset. A tempo estimation algorithm (Zapata & Gomez, 2011) was presented in the music information retrieval evaluation exchange (MIREX 2011) in the Tempo estimation task. The algorithm used a heuristic decision tree strategy to analyze the relations between the tempo estimations, and obtain the perceptual slow and the fast tempo from a audio song.
Due to implementation limitations (different language implementation) this combination algorithm uses four tempo estimation approaches (BeatIt, Davies, Ellis and MPEG7-XM) and the combination algorithm steps are:

1. Sort all the tempo estimation values and eliminate the repeated ones.
2. Cluster tempo values with differences of 4% (e.g: 127.6, 125.8 , 128.4) and calculated the median value of each cluster.
3. Check if each value has a relation of ($\frac{1}{2}$, 2, $\frac{1}{3}$ or 3) and eliminate those not related.
4. Heuristic
   - If only two values are obtained, the lowest value is the slow tempo (T1) and the highest value is the fast tempo (T2). The strength of T1 relative to T2 is taken from the Ellis Results.
   - If there are 3 values with a binary relation between them, the lowest value is the slow tempo (T1)and the double value is the fast tempo(T2). The strength of T1 relative to T2 is used from the Ellis Results.
   - If there are 3 values and two of these had a ternary relation between them, we take these two numbers and the lowest value is the slow tempo (T1) and the double value is the fast tempo(T2). The strength of T1 relative to T2 is taken from the Ellis Results.
   - If there are more than 3 values, we choosed the values related with the BeatIt estimation, then the heuristic rules were checked to obtain the slow tempo (T1), the fast tempo(T2) and the strength of T1 relative to T2, taken from the Ellis Results.
5. Output: slow tempo(T1), fast tempo(T2) and the strength of T1 relative to T2 values between [0-1].

2.3.2. MIREX Tempo task evaluation results

The MIREX 2011[15], 2010 [16] and 2006 [17]: Audio Tempo Extraction - MIREX06 Dataset results are presented in Table 2.6, sorted by Tempo P-score per year. Our algorithm, **ZG1**, obtained the second best mean results (2011) in both tempi correct value (0.5714) and the third overall position until 2011.

Table 2.6: Results MIREX 2011, 2010 and 2006: Audio Tempo Extraction

Year	Algorithm	Tempo P-Score	One tempo correct	Both tempi correct
2011	GKC3	**0.8290**	0.9429	**0.6214**
	FW2	0.7385	0.8357	0.5429
	ZG1	0.7275	0.8214	0.5714
	SP1	0.7105	0.9286	0.3857
	GKC6	0.6777	0.8214	0.4286
	SB5	0.6559	0.8429	0.3500
2010	GKC1	0.8099	**0.9643**	0.5000
	NW2	0.7875	0.9143	0.5000
	ES1	0.7714	0.9071	0.5500
	TL1	0.7639	0.8929	0.4786
	BES2	0.7429	0.9143	0.4857
	OL1	0.6679	0.8786	0.3786
	GT1	0.6150	0.6929	0.5071
2006	Klapuri	0.806	0.9429	0.6143
	Davies	0.776	0.9286	0.4571
	Alonso	0.7242	0.8929	0.4357
	Alonso	0.6931	0.8571	0.4571
	Ellis	0.673	0.7929	0.4286
	Antonopoulos	0.669	0.8429	0.4786
	Brossier	0.62	0.7857	0.5071

The GKC3 approach by Gkiokas et al. (2011) has the highest reported accuracy on the audio tempo task on MIREX until 2011. This method extracts two main feature classes using percussive/harmonic separation of the audio signal, in order to extract energy in the percussive signal and chroma in the harmonic signal. Periodicity analysis is carried out by a bank of resonators as in the others approaches with similar results from section 2.2.3, suggesting that the improving of the algorithm accuracy lies in good a feature extraction.

[15] http://nema.lis.illinois.edu/nema_out/mirex2011/results/ate/
[16] http://nema.lis.illinois.edu/nema_out/mirex2010/results/ate/
[17] http://www.music-ir.org/mirex/wiki/2006:Audio_Tempo_Extraction_Results

The proposed method uses Ellis and Davies tempo algorithms presented in MIREX 2006 in the same dataset. When compared ZG1 with both algorithms, the results of ZG1 are higher for the both tempi correct value. Finally, the ZG1 algorithm had better results in at least one tempo correct value compared with the Ellis result. These results could be improved using a combination of more algorithms with better performance.

2.4. Discussion and future Work

2.4.1. The dataset and the metric

The ISMIR 2004 tempo dataset is used to compare the new approaches in tempo estimation with the previous published research and for 91.18% of the song-excerpts in the dataset, based on the evaluation results, their metrical levels can be estimated with the estimations of all the algorithms and the double is the most common error, followed by the $\frac{4}{3}$ error. In order to obtain more information about the data of all these results, future work could include annotation data of the meter, *tactus* and *tatum* of each song-excerpt. This information helps to improve the analysis of the results and errors inherent in the metric 1 and 2, because its not clear if the relations 2, $\frac{1}{2}$, 3 and $\frac{1}{3}$ are a metrical level error or another kind of error. As a consequence, the MIREX evaluation method could be used in this dataset, if the information of the metrical levels are added.
Some song-excerpts of the dataset are not tempo stable, therefore information of the tempo stability of the song (if the song has a bimodal tempo or a very variable tempo) will be useful for performance analysis of the algorithms. To detect the slow or fast tendency of the algorithms and a flat tempo distribution of the dataset, more data will be needed. Moreover, more types of music genres need to be included to have a better tempo estimation analysis per genre.

2.4.2. Performances by genre and limitations

Comparing the evaluation results againts the past evaluation in 2004, classical music still remains less accurate for metric 1 and 2 and the academic approaches (Klapuri, BeatIt, Ellis, Qmtempo, IBT) compared with most of the commercial approaches had better performance in most of music genres even for electronic and percussive music. The commercial Auftakt algorithm had a statistical performance equal to Klapuri and BeatIt for the metric 1, but for metric 2 the algorithms had a statistical significance difference in their performances. Working with Percussive music such as jazz the algorithms had difficulties determining the metrical level, because of their complex rhythms. As past results, most of the approaches had good accuracy in percussive genres as latin and afrobeat.

A robust method capable to estimating the tempo in classical music with soft onsets or soft transitions does not exist. For most of the music with strong emphasis on singing voice (e.g: fado, greek) tempo estimators fail to detect the *tactus* metrical level. This problem is addressed in Chapter 4. Based on the results, a limitation of all of the algorithms is to estimate lower tempo (49 BPM or less, in the evaluated dataset) in the *tactus* metrical level, but this problem be caused by the window size in the feature extraction or the periodicity function calculation, or by also an effect of the tempo correction in some algorithms.
Furthermore, most of the errors are due to double tempo estimation, ternary meters and low tempi, which reflects the current glass ceiling in tempo estimation. Research should be devoted to metrical level estimations, binary and ternary detection, soft beats detection and slow tempo estimation.

2.4.3. Challenges in the design of an algorithm for tempo estimation

Designing an unique system to estimate tempo for all genres and tempo range is one of the goals in automatic rhythm description. After analyzing the best performing algorithms in this tempo estimation evaluation, we found these common characteristics: frequency band decomposition, periodicity detection prior to the weighting of each band and multi-band integration; detection of the metrical levels (*tatum*, *tactus* and musical measure) along with a tempo correction function in order to reduce the number of double and half tempo estimations.
Nevertheless the evidence does not show which audio features are better for tempo estimation (e.g. spectral flux, energy envelope, energy changes) along with the pulse induction methods used in different approaches (e.g. ACF, AlonsoSP = spectral product, Scheirer = comb filter bank), which had statistically similar performance. It is not clear which periodicity function (e.g. ACF, bankcomb filters, DFT, spectral product) or combination of these functions are appropriate for pulse induction.
Most of the tested tempo estimators use ACF to find periodicities in the signal; however, their performances are statistically different. In addition, different methods to select the best tempo in the periodicity signal are used (e.g. histogram, peak selection, tempo hypothesis with agents setup, Viterbi decoding and clustering); however, it is indistinguishable, through the results, which of these methods yields better results by itself. For future work, an analysis of the combination of different periodicity functions and different selection methods would be needed.

The evaluation results shows that the algorithms with better performance on this dataset use band decomposition as a part of their signal processing, though it is not clear which band decomposition (e.g 5, 6, 36 bands and linear, logarithmic or perceptually spaced) is more suitable for rhythm description. As future work, tempo estimation could be improved by identifying which periodicity function, audio features and which band decomposition method its better.
The best performing algorithms (Klapuri, Hyb2 and BeatIt) are based on the following steps: frequency decomposition, periodicity detection prior to the multi-band integration, *tatum* detection and a post-processing block which reduces the number of double and half error tempo estimations. Klapuri's algorithm is still the best performing one among all the evaluated algorithms but its performance can be improved by a combination with other tempo estimation methods.
Finally, the computational global tempo estimation is intrinsically linked to determining beat positions, and most of the best tempo estimators in this evaluation are beat trackers, so an enhancement of beat-tracking systems would shed light on this relation for tempo estimation improvement.

2.4.4. Combination of algorithms for tempo estimation

Despite the good results of the combination method, in practice, this algorithm requires considerable effort. To install appropriate system components and operating systems to make all of the algorithm work. Additionally, the algorithm only analyzes the binary and ternary relations between the algorithms results, the results could be improved by searching for odd or changing meter and for future work, an implementation of a single system that unifies different tempo estimation approaches could be done. For example, by using the methodology of query by committee concept proposed in chapter 3.
Finally, two open questions remains about how their results could be integrated to increase tempo estimation accuracy and how can be combined the results of the tempo systems, to estimate the slower (*tactus*) and faster(*tatum*) tempo of a song.

2.5. Summary

In this chapter, we evaluated and compared 32 state-of-the-art algorithms for tempo estimation, that we consider representative of current approaches. We observed that current accuracy levels are around 91% on the considered dataset and the best result, obtained by a beat tracker, can be enhanced by a heuristic decision tree strategy combination with the other methods and their results are better than each approach by itself. Although, an open question remains about how their results could be integrated to improve tempo estimation to get better performance.
The best performing algorithms share the following characteristics : frequency decomposition, periodicity detection prior to the multi-band integration, *tatum* detection and a post-processing block which reduces the number of double and half error tempo estimations. Moreover, according to the statistical significance analysis of the evaluation results, we conclude that among the tested algorithms involving band decomposition, those computing the periodicity detection before the multi-band integration and beat estimation achieve better results. While the accuracy of our tempo estimation system is not yet the best tempo algorithm, we propose a method for improving performance which is specific to our goal of combining multiple systems. Consequently, these analysis became the motivation to investigate the limitations and problems in beat tracking in order to improve automatic rhythm description.

CHAPTER 3

Beat Tracking

The automatic extraction of beat times from music signals is a mature research topic within music information retrieval (MIR) and is a key aspect of computational rhythm description (Gouyon & Dixon, 2005). The aim of a beat tracking system is to recover a sequence of time instants consistent with how a human might tap their foot in time to music. For a recent review see (Degara, 2011, ch.2). Beat tracking systems are now considered "standard" processing components within many MIR applications, such as chord detection (Mauch et al., 2009), structural segmentation (Levy & Sandler, 2008), cover song detection (Ravuri & Ellis, 2010), automatic remixing (Hockman et al., 2008) and interactive music systems (Robertson & Plumbley, 2007), see Section 1.3.

While the efficacy of beat tracking systems can be evaluated in terms of their success of these end-applications, e.g. by measuring chord detection accuracy, considerable effort has been made to on the evaluation of the beat tracking systems directly through the use of annotated test databases in particular within the MIREX initiative. In the small number of comparative studies of automatic beat tracking algorithms with human tappers (Collins, 2006; Davies & Plumbley, 2007; Holzapfel et al., 2012b; McKinney et al., 2007; Scheirer, 1998) musically trained individuals are generally shown to be more adept at tapping the beat than the best computational systems. Given this gap between human performance and computational beat trackers, we consider there is room for improvement.

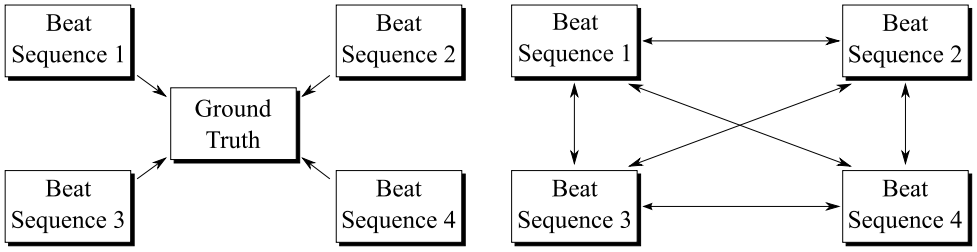
In order to devise a method for beat tracking using a combination of different approaches, in machine learning, *selective sampling* approaches have been proposed to select informative samples the absence of ground truth (Dagan & Engelson, 1995). In this work, we follow the Query by Committee concept by Seung et al. (1992) which assign a degree of agreement to a given piece by measuring the mean mutual agreement (*MMA*) between a set of state of the art beat tracking approaches. In fact, we consider that the beat estimation

This section is based upon work in collaboration with Andre Holzapfel, Matthew E. P. Davies, João Lobato Oliveira and Fabien Gouyon. This is a compilation of papers published in a journal and peer reviewed conferences, Holzapfel et al. (2012a,b); Zapata et al. (2012b)

that most agrees among the others is the most reliable, and the song in question will be difficult in case of not consensus among beat estimations. When assembling our committee of beat trackers, we take into account that the committee should be characterized by both high accuracy and diversity (Melville & Mooney, 2004). Similar concepts have been evaluated in the domain of speech processing by Dagan & Engelson (1995). Mandel et al. (2006) develop an approach which includes user interaction to identify informative samples for training a music retrieval system. However, to our knowledge, *selective sampling* has not yet been applied to evaluate music signal processing applications like beat tracking.
The remainder of this chapter is structured as follows; In Section 3.1, we use the mutual agreement to address issues of evaluation measures and the choice of beat tracking algorithms for mutual agreement computation. In Section 3.2, we use an existing beat tracking database to determine system parameters for the *MMA* computation, and demonstrate the validity of our approach. Based on the previous sections, in Section 3.3.1 we give an overview of a stand alone beat tracking method based on mutual agreement and a committee conformed by multiple onset detection functions. In Section 3.4 we describe the experimental setup used to select the best committee members. In Section 3.5 we present the performance and behavior of each onset detection function and demonstrate the improvement of the system on a manually annotated dataset. Finally, in Section 3.6 we present the discussions of the results and areas for future work.

3.1. Mutual sequence agreement

Our approach is motivated by the Query by committee concept proposed by Seung et al. (1992), which provides a method for selecting informative data samples by measuring the agreement between the committee members. Even though most beat tracking systems are optimized manually, we can compare this optimization process with a learning process, and the current state of the art can be considered a committee of learners that can profit from selecting the committee member which most agree with the others.
A graphical representation for estimating the committee agreement of a music sample for beat tracking when ground truth is given is shown in Figure 3.1a. Here, a set of N beat sequences is calculated for a given sample using N different beat trackers. These beat sequences are then compared with the given ground truth of the piece using an evaluation measure, and the *mean ground truth performance* of all beat trackers, **BT-MGP**, on this piece can serve as an estimate of its committee agreement. Note that this is different from calculating the mean ground truth performance of a single beat tracker over a whole data set, which can serve as an indicator of its performance and will be referred to as **D-MGP** throughout the chapter.



(a) Ground truth given (b) No ground truth

Figure 3.1: Setups for determining difficulty of a sample for $N = 4$ beat trackers, (a) with and, (b) without ground truth.

However, when no ground truth is given, an unknown sample might be labeled as "interesting" for beat tracking if a committee of beat trackers disagrees in their estimates of the beat. Hence, the beat sequences of the N beat trackers are compared to each other, creating a complete graph with $N(N-1)/2$ mutual agreement values on its edges, as shown in Figure 3.1b. The mean weight of the edges is equal to the mean mutual agreement between the beat sequences, **BT-MMA**, which can then serve to estimate for the level of agreement of a piece for beat tracking and to select the committee member that more agree with the other ones. There are two important questions that have to be considered in order to use the method of committee for beat tracking.

1. Which evaluation measure to apply to compute the mutual agreements?

2. Which beat trackers to compare?

3.1.1. Evaluation measures

Our mutual agreement measure relies on the use of an objective beat tracking evaluation method to determine the relationship between pairs of beat sequences. The selection of this evaluation method poses a problem since there is no commonly accepted technique for measuring beat tracking performance. This lack of consensus has led to many approaches being developed, each with differing parameters and/or methodologies. For a review and further discussion, see Davies et al. (2009a). The variations among evaluation methods[1] arise due to differing hypotheses on how to address the localization between beat times and annotations (e.g. by the use of tolerance windows), and how to contend with ambiguity over the validity of metrically related sequences. The eventual choice of a specific evaluation method is usually made in the context of a particular application. For example, when evaluating a real-time

[1] All measures were computed using the beat tracking evaluation toolbox, available at http://code.soundsoftware.ac.uk/projects/beat-evaluation

beat tracking system, a continuous relationship between beats and annotations may be an important criterion (Stark et al., 2009). Or, for chord recognition, permitting many different interpretations of the beat may be detrimental to chord detection accuracy (Bello, 2007) hence it may be advisable to restrict the range of alternate interpretations of the beat.

Our reason for using a beat tracking evaluation method is somewhat different, since our primary interest is not in identifying where beat sequences agree with each other per se, but rather in finding the one which most agree. While this agreement could be measured in terms of ambiguity in metrical level or beat phase, this is of limited use since these beat sequences could be considered "somehow" related. Of greater importance for our application is finding when the beat sequences are completely related or unrelated. This is based on our intuition that beat trackers are usually built out of similar components, and the lack of consensus of their outputs is the condition that we are interested in to find the beat tracker that more agrees with the others. Based on this reasoning, the choice of evaluation method may appear trivial, since we could simply look for cases where the evaluation score is high but we need to know when the beat sequences don't agree or their value was close to 0% for any evaluation method. To explore this hypothesis further we briefly address the properties of three evaluation methods which cover the main types of techniques currently used. For each we describe its basic functionality and indicate the conditions under which a minimal accuracy score can occur.

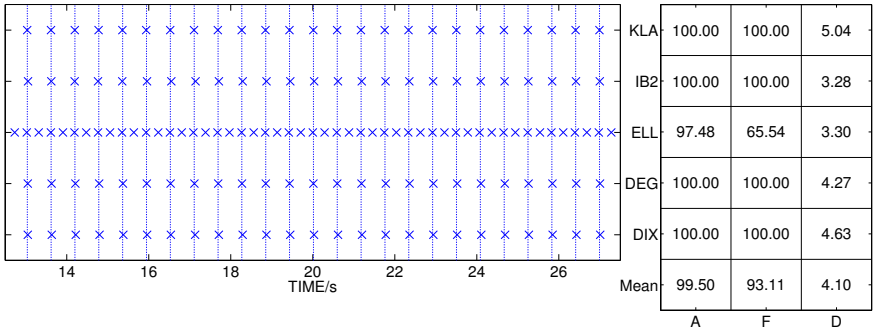
F-measure (Dixon, 2007): Beats are considered accurate if they fall within a $\pm$70ms tolerance window around annotations. Accuracy in the range from [0 -100]% is measured as a function of the number of true positives, false positives and false negatives. If the beat sequences are tapped at metrical levels related by a factor of two (but otherwise well aligned), this causes the score to drop from 100% to 66.7%. A score of 0% can only occur if no beat times fall within any tolerance windows. The most likely scenario for this score is if the beat sequences tapped in anti-phase (i.e. on the "off-beat"). Completely unrelated beat sequences typically score around 25% (Davies et al., 2009a) by virtue of beats arbitrarily falling within the range of tolerance windows.

AMLt (Hainsworth & Macleod, 2004): A continuity-based method, where beats are accurate when consecutive beats fall within tempo-dependent tolerance windows around successive annotations. Beat sequences are also accurate if the beats occur on the off-beat, or are tapped at double or half the annotated tempo. The range of values for AMLt is[0 -100]%. A score of 0% can only occur if no two consecutive beats fall within the specified tolerance windows; this is most likely the result of the beat sequences being related by an unspecified metrical relationship, e.g. "2 against 3" (Davies et al., 2011). As with F-measure, unrelated sequences do not score 0%, being closer to 18% (Davies et al., 2009a).

Information Gain (Davies et al., 2011): Accuracy is determined by calculating the timing errors between an annotation and all beat estimations within a one-beat length window around the annotation. Then, a beat error histogram is formed from the resulting timing error sequence. A numerical score is derived by measuring the K-L divergence between the observed error histogram and the uniform case. This method gives a measure of how much information the beats provide about the annotations. The range of values for the Information Gain is 0 bits to approximately 5.3 bits, where the upper limit is $\log_2(K)$ for K histogram bins. Maximal Information Gain is the result of all beat error measurements falling within a single histogram bin, hence the choice of K is important and should be neither too large nor too small; $K = 40$ histogram bins is an appropriate choice. An information gain of 0 bits is obtained, in the limit, when the beat error histogram is uniform, i.e. where the beat sequences are totally unrelated.
To illustrate the differences in beat tracking outputs and the effect of different evaluation methods we examine two examples, Figure 3.2 shows beat annotations and estimations for two song excerpts. It is evident that in Figure 3.2a beat estimations widely agree, apart from a tempo doubling by the ELL algorithm. This doubling is penalized by the F-measure by a value close to 66%, as explained above. All three measures are characterized by high mean ground truth performances for this song. For the example shown in 3.2b all beat estimations disagree both mutually and also with the ground truth. However, both F-measure and AMLt result in mean accuracies of about 35%, while only Information Gain (D) is characterized by a value close to zero.
Based on the properties of these evaluation methods, all the evaluation measures can detect when all the beat sequences agree, but the Information Gain approach would appear most suited to our purpose since it is the only method guaranteed to be close to 0 only in the condition where the beat sequences have no meaningful relationship. However, to confirm this we retain all three evaluation methods throughout the subsequent analysis. In our notation, we will add a subscript $z \in \{F, A, D\}$ for F-measure, AMLt and Information Gain, respectively, whenever a distinction is of importance (*e.g.* BT-MMA$_D$ for BT-MMA using Information Gain).

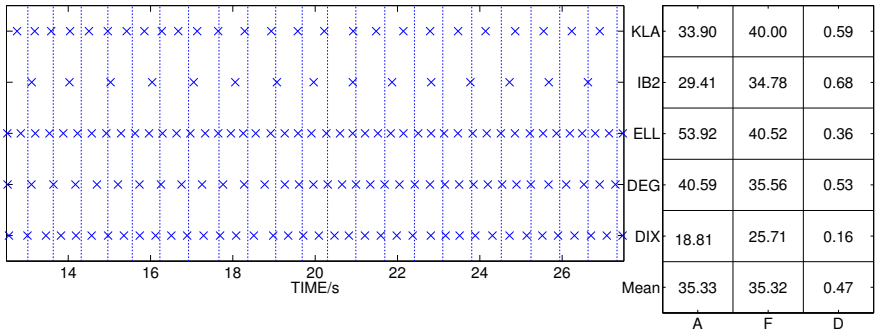
3.1.2. Choice of committee members

Implementations of various beat tracking algorithms were collected including those freely available online and others kindly provided by the authors of the systems on request. In total we compiled an initial committee of 16 beat trackers (Table 3.1), describing their principal characteristic like Figure 1.3. In practice, to use all the collected beat trackers requires considerable effort, such as installing appropriate system components and operating systems necessary to make all of the algorithms run. Furthermore there was both considerable variability in the computational complexity of the algorithms, with some al-



	A	F	D
KLA	100.00	100.00	5.04
IB2	100.00	100.00	3.28
ELL	97.48	65.54	3.30
DEG	100.00	100.00	4.27
DIX	100.00	100.00	4.63
Mean	99.50	93.11	4.10

(a) Example for an easy song (Busta Rhymes)



	A	F	D
KLA	33.90	40.00	0.59
IB2	29.41	34.78	0.68
ELL	53.92	40.52	0.36
DEG	40.59	35.56	0.53
DIX	18.81	25.71	0.16
Mean	35.33	35.32	0.47

(b) Example for a difficult song (Tom Waits)

Figure 3.2: Ground truth annotations shown as dotted vertical lines. Beat estimations for five algorithms (see Table 3.1) are superimposed as crosses. The tables list the performances of the evaluation methods for each song, and their means.

gorithms slower than the fastest by up to two orders of magnitude, and large variation in beat tracking performance (see Section 3.2). So as to make the results of this work more easily reproducible we propose a method to select a subset of these algorithms. The selected algorithms should be characterized by good performance, but at the same time care should be taken to include approaches that complement each other. The goal is to obtain a small but diverse committee, where each beat tracking implementation is publicly available and not too demanding in terms of execution time. To find a subset of the $N = 16$ beat tracking algorithms we make use of oracle scores (the best performance per song), following the sequential forward selection (SFS) method, computed as depicted in Algorithm 2. It requires a set of data with available beat annotations, with a number of samples equal to N_S. It starts with computing the ground truth performance vectors of length N_S for all $i = 1...N$ beat trackers on this dataset, D-GP(i). We start by including the best single performing algorithm into the committee (first pass through the While loop).

Table 3.1: Summary beat tracking approaches. Tracking system: ACF - Autocorrelation function, CFB - Comb filter bank, DP - Dynamic programming, IOI - inter onset interval, BLSTM - bidirectional Long Short-Term Memory neural network, MA - Multiple agent system, PF - Particle Filtering, STFT - short time fourier transform

Beat Tracker	**Reference**	**Onset Detection Function**	**Tracking System**	**Language**	**Real Time**
Aubio (AUB)	Brossier (2006)	Complex spectral difference	ACF	C/C++	No
Beatit (BIT)	Bonada & Gouyon (2006)	Energy Flux	ACF	C/C++	No
Beatroot (DIX)	Dixon (2007)	Spectral flux	IOI, MA	Java	No
BeatUJAEN (BUJ)	Mata-Campos et al. (2010)	Sinusoidal perceptual model	IOI, MA	Matlab	No
Boeck (BOE)	Böck & Schedl (2011)	Logarithmic Mel spectrogram	BLSTM, ACF	Python	No
BpmHistogram (BH)	Aylon & Wack (2010)	Novelty function	STFT	C/C++	No
Davies (DAV)	Davies & Plumbley (2007)	Complex spectral difference	ACF, CFB, DP	Matlab	No
Degara (DEG)	Degara et al. (2012)	Complex spectral difference	ACF, HMM	Matlab	No
Ellis (ELL)	Ellis (2007)	Mel auditory feature	ACF, DP	Matlab	No
Hainsworth (HAI)	Hainsworth & Macleod (2004)	Harmonic feature	PF	Matlab	No
IBT (IB1)	Oliveira et al. (2010)	Complex spectral difference	MA	C/C++	Yes
IBT off-line (IB2)	Oliveira et al. (2010)	Complex spectral difference	MA	C/C++	No
Klapuri (KLA)	Klapuri et al. (2006)	Bandwise Accent signal	CFB, HMM	Matlab	No
Lee (LEE)	Lee (2010)	Mel auditory feature	STFT, DP	Matlab	No
Scheirer (SCH)	Scheirer (1998)	Temporal envelope difference	CFB	C/C++	No
Stark (STA)	Stark et al. (2009)	Complex spectral difference	ACF, CFB, DP	Matlab	Yes

The next pass through the while loop combines it with each other beat tracker and obtains an *oracle* vector of each pair of beat trackers. The best combination is selected and a vector is built that contains the oracle performance of these two beat trackers for each file. This procedure is repeated until all beat trackers are included in the subset. We can then look at which order the algorithms entered the subset and what improvement in performance was achieved by their inclusion. A choice of beat trackers guided by this algorithm takes into account both accuracy and diversity.

Algorithm 2 Sequential forward selection (SFS), Oracle score computation

init: $oraclegp = \text{zeros}(N_S, 1)$, $oracle = \{\}$.
for $i = \{1, ..., N\}$ **do**
  Compute D-GP(i)
end for

while $length(oracle) \leq N$ **do**
  for All i not in $oracle$ **do**
    $temp(i) =$ pairwise max($oraclegp$,D-GP(i))
  end for
  add $\hat{i} = \arg\max_i \text{mean}(temp(i))$ to $oracle$
  $oraclegp = temp(\hat{i})$
end while

3.2. Beat Tracking Evaluation and Applying *MMA* to an existing dataset

The largest dataset for beat tracking evaluation to date is a compilation of 4 beat annotated datasets [CUIDADO (70 songs excerpts), Hainsworth (221 songs excerpts), Klapuri (474 songs excerpts), and SIMAC (595 songs excerpts)] introduced by Gouyon (2005), and it contains a total of 1360 excerpts from different styles of music. The dataset will be referred to as **Dataset1360** throughout the following sections. Original genres were grouped in 10 classes, mostly with respect to their instrumental or rhythmic contents (see Table 3.2).

Using this dataset we evaluated the accuracy and the diversity of the available 16 beat trackers. Based on these results we will define our committee of beat trackers and give a proof of concept for our *MMA* method to improve automatic beat tracking and to determine the most appropriate evaluation method.

Table 3.2: Genre Distribution of the Dataset1360

Genre	Songs	Description
Acoustic	84	Mostly sung pieces accompanied with acoustic instrument, as the guitar, some percussion but no loud drums.
Jazz/Blues	194	Quite heterogeneous set, many instrumental pieces with lots of horns, jazz-like drum playing.
Classical	204	Mostly orchestral music, symphonies or sonatas, few operas.
Classical solo	79	Solo instruments such as piano, guitar or organ.
Choral	21	Just Choirs.
Electronic	165	lots of electronic drums, mostly strong beats.
Afro-American	93	$\frac{4}{4}$ time signatures and clear drum patterns on the great majority of excerpts.
Rock/Pop	334	Quite heterogeneous set, mostly sung pieces, with drums.
Balkan/Greek	144	Sung pieces accompanied by acoustic instruments, typical folklore music from Greece and Balkans.
Samba	42	Sung pieces accompanied by acoustic instruments, with typical second and fourth beats marked by percussion, folklore music from Brazil.

3.2.1. Accuracies of potential committee members

In Table 3.3 the D-MGP of all 16 beat trackers are given for Dataset1360. In order to compare the beat trackers, a one-way ANOVA followed by a series of t-tests with level of significance of $\alpha = 0.05$ was performed. Tukey's HSD adjustment was used to account for the effect of multiple comparisons. The best beat tracking results without statistically significant differences are depicted in boldface.

It can be seen from Table 3.3 that a subset of beat trackers perform significantly better than most of the others. The set of the best beat trackers varies slightly depending on the evaluation measure which is applied. Comparing the D-MGP values of the approaches with the mean values of all beat trackers we can see that some approaches perform worse than the mean for all evaluation measures. When looking to finding a subset of committee members we considered the need for accuracy in beat tracking, since poorly performing beat trackers can lead to an over-estimation of disagreement and is better to select our output between the best committee members.

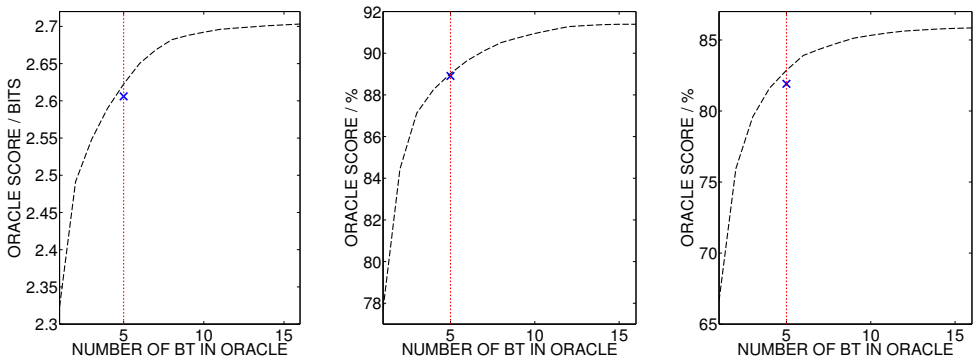
Table 3.3: Mean ground truth performance of each BT (D-MGP) on **Dataset1360**. Bold numbers indicate best performances.

Beat Tracker	AMLt (%)	F-measure (%)	Inf. Gain (bits)
Aubio (AUB)	50.6	49.4	1.58
Beatit (BIT)	61.0	52.7	1.62
Beatroot (DIX)	70.8	61.7	1.98
BeatUJAEN (BUJ)	41.6	33.9	1.18
Boeck (BOE)	58.7	**66.6**	1.98
BpmHistogram (BH)	57.3	51.7	1.43
Davies (DAV)	**75.9**	**62.8**	**2.25**
Degara (DEG)	**77.7**	**65.3**	**2.26**
Ellis (ELL)	60.0	55.1	1.76
Hainsworth (HAI)	59.6	51.1	1.84
IBT causal (IB1)	58.0	55.2	1.67
IBT non-causal (IB2)	**73.8**	60.5	1.92
Klapuri (KLA)	**77.7**	**65.5**	**2.32**
Lee (LEE)	26.4	48.8	1.09
Scheirer (SCH)	49.0	56.2	1.69
Stark (STA)	71.0	59.5	2.03
Mean	60.6	56.0	1.79

3.2.2. Selecting the committee

We now illustrate the effect of choosing the committee members based on oracle performances as described in Section 3.1.2. The development of the oracle scores are depicted in Figure 3.3. A saturation effect can be observed when the number of beat trackers in the subset increases, and we decided to limit the number of beat trackers to 5 (dotted red line). The order in which algorithms entered the oracle slightly varied between the evaluation measures. We decided to choose the 5 beat trackers based on their average ranking obtained from the three evaluation measures, which gave [KLA, DEG, HAI, BOE, IB2]. This ranking results have higher diversity of approaches instead of ordering according to ground truth performance. For example, the DAV[2] algorithm is not among the best five in the ranking. This is caused by the fact that the DAV and DEG algorithms are very similar, and DAV does not improved to much after DEG entered. Instead of the DAV algorithm, the HAI and the BOE algorithm enter the committee, which follow quite different approaches from the KLA and the DEG algorithm and seem to increase the diversity of the committee. However, we chose not to include the approaches by Böck & Schedl (2011) (BOE) and Hainsworth & Macleod (2004) (HAI) into

[2]Note, we use an improved version of the original algorithm which is implemented as a Sonic Visualiser plugin.



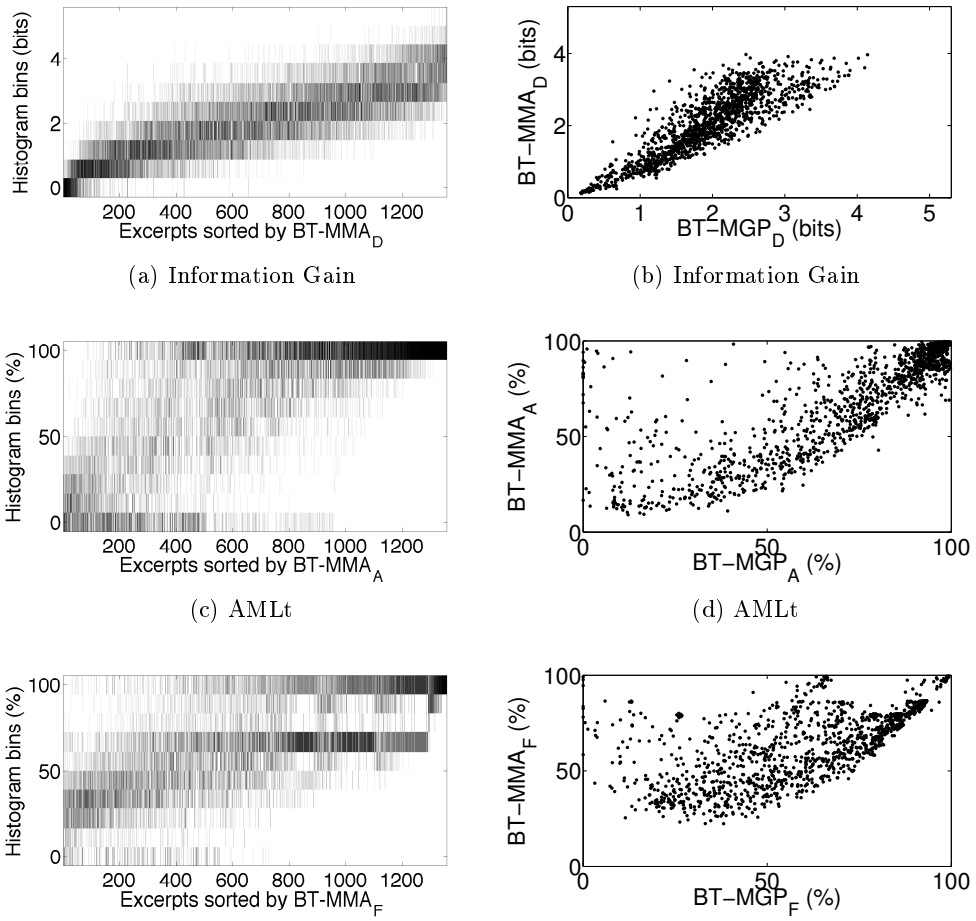
(a) Information Gain (b) AMLt (c) F-measure

Figure 3.3: Development of the oracle scores for the three evaluation measures. Performance of the chosen committee is depicted by a blue cross.

our committee of 5 beat trackers for reasons of portability, computation time and public availability. Instead, we included the widely available approaches by Dixon (2007) (DIX) and Ellis (2007) (ELL). This lead to non-significant decrease in oracle performance (marked by a cross in Figure 3.3) by 0.63%, 0.13% and 1.15% for Information Gain, AMLt, and F-measure, respectively. Finally, to form our committee we selected five state of the art and publicly available beat trackers: Dixon (Dix.) (Dixon, 2007), Degara (Deg.) (Degara et al., 2012), Ellis (Ell.) (Ellis, 2007), IBT (Oliveira et al., 2012), and Klapuri (Kla.) (Klapuri et al., 2006). These systems had the performance and diversity necessary to compute a reliable *MMA*. We hope that the chosen committee will enable other researchers to most easily reproduce results presented in this work.

3.2.3. Mean Mutual Agreement (*MMA*)

After the selection of committee members, mutual agreement between the sequences obtained from the 5 beat trackers were computed using the three evaluation measures described in Section 3.1.1. Then, for each evaluation measure, mutual agreements for a particular piece were summarized in a mutual agreement histogram with 11 bins spanning the whole range of values of the particular evaluation measure (*e.g.* 0% to 100% for AMLt). In the left column of Figure 3.4 these histograms are depicted for **Dataset1360**. The histograms are sorted by their BT-MMA value for each evaluation method. Dark colors in the histogram plots indicate a high population of the specific histogram bin. In the right column of Figures 3.4, scatter plots of the BT-MMA over the mean ground truth performance BT-MGP are shown. For our purposes, BT-MMA can predict BT-MGP at least for difficult pieces. These are located at low BT-MGP values, while easier pieces are found at higher BT-MGP values, *i.e.* in



(a) Information Gain (b) Information Gain

(c) AMLt (d) AMLt

(e) F-measure (f) F-measure

Figure 3.4: Left side: Each column depicts the histogram obtained from the $5 * 4/2$ mutual agreements of the beat sequences for a song in Dataset1360, histograms are sorted by their mean values (BT-MMA). Dark colors indicate high histogram values. Right side: *MMA* versus MGP scatter plots.

the region where beat trackers perform well in the mean for a specific sample. Comparing the scatter plots for the three evaluation measures we can observe that the BT-MMA$_D$ in Figure 3.4b is characterized by the highest correlation with the BT-MGP. This correlation is even larger for low BT-MMA$_D$ values, which indicates that low BT-MMA$_D$ can reliably predict low ground truth performance. The other two scatter plots (Figures 3.4d and 3.4f) show an increased correlation only for high ground truth performance, *i.e.* in the upper right corner of these scatter plots. For lower BT-MGP, F-measure in particular cannot be applied to predict BT-MGP using BT-MMA. This difference of

Information Gain on the one side and F-measure and AMLt on the other can be attributed to the property of Information Gain of having an unambiguous "true" zero value, as explained in Section 3.1.1.
By observing the histogram plots in the left column of Figure 3.4, it is apparent that only for the Information Gain in Figure 3.4a a continuous transition from histograms centered at low values to histograms centered at high values exists. The other two measures are characterized by generally flatter histograms, and the F-measure histograms are often characterized by simultaneous high values for 100% and 66.7%. This can be ascribed to sequences that are at metrical levels related by a factor of 2 (see 3.1.1). These characteristics imply that the computation of a mean is most reliable for the Information Gain. Hence, we conclude that using Information Gain for *MMA* computation is superior to using either F-measure or AMLt.

3.2.4. Measuring Mutual Agreement, Mean Mutual Agreement

The measurement of Mean Mutual Agreement (*MMA*) gives an agreement level of the beat trackers per song in a dataset based on the mutual (dis-)agreement between a designated committee of learners. As depicted in Figure 3.5, the *MMA* is computed using the beat outputs (or beat sequences) of a committee of N beat trackers on a musical piece, by measuring the mutual agreement $MA_{i,j}$ (see eq. (3.4)) between every pair of estimated beat tracker outputs i and j, and retrieving the mean of all $N(N-1)/2$ mutual agreements:

$$MMA = \frac{1}{N(N-1)/2} * \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} MA_{i,j}. \tag{3.1}$$

In addition to calculating the *MMA* as a summary statistic, we can easily identify the mutual agreement, MA_i, of the beat tracker output i which most agrees with the remainder of the committee, the *MaxMA*, and the beat tracker output i which agrees the least, the MinMA :

$$MA_i = \frac{1}{N-1} * \sum_{j=1, j\neq i}^{N} MA_{i,j}, \tag{3.2}$$

$$\begin{cases} MaxMA = max_i\left(MA_i\right) \\ MinMA = min_i\left(MA_i\right) \qquad i,j = 1,\ldots,N \wedge i \neq j \end{cases} \tag{3.3}$$

In order to measure the mutual agreement $MA_{i,j}$ between each pair $\{i,j\}$ of beat tracker outputs, a beat tracking evaluation method must be chosen. In 3.1.1 we reviewed the properties of existing evaluation methods (Davies et al., 2009a) and selected the Information Gain approach by Davies et al. (2011) (InfGain) as the only one with a true zero value, able to match low *MMA*

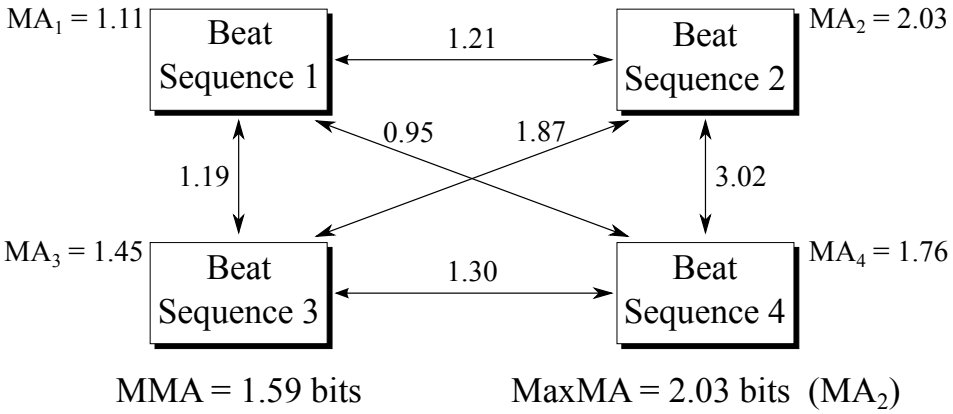


Figure 3.5: Example calculation of the *MMA* and *MaxMA* for a song with the beats estimated from a committee of four beat trackers.

(measured in bits) with unrelated beat tracker outputs:

$$MA_{i,j} = InfGain(i,j), \qquad i,j = 1,\ldots,N \wedge i \neq j. \tag{3.4}$$

The Information Gain measure is determined by forming a beat error histogram representing the timing error between beat sequences. A numerical score is calculated as a function of the entropy of the histogram. The range of values for the InfGain is 0 bits to approximately 5.3 bits, where the upper limit is $\log_2(K)$ for K=40 histogram bins. For more details see Davies et al. (2011).

3.2.5. Maximum Mutual Agreement (*MaxMA*)

In this experiment we used our committee of beat trackers (Selected in Section 3.2.2) on **Dataset1360** and calculated BT-MMA$_D$ for each sample. We then excluded those samples with BT-MMA$_D$ below a specified threshold. The threshold was incremented in steps of 0.3 bits from 0 to 3 bits. We now tried to select the beat sequence with maximum mutual agreement with the committee, which we denote, ***MaxMA*** (Equation 3.3). For each sample (at a given threshold), we simply selected the beat sequence with the maximum mutual agreement (*MaxMA*) with the other four sequences as the most reliable beat estimation. In effect we assume that the beat tracker that best agrees with the rest of the committee is the most reliable algorithm. In Figure 3.6, we compare the *MaxMA* approach to another viable option, that of picking the beat tracker (Klapuri et al., 2006) with the best mean overall performance from our experiments in Section 3.2. We denote this option *Best mean*. To illustrate the upper limit on performance we also include the theoretical optimum **Oracle** (that picks the most accurate beat tracker for each individual sample).

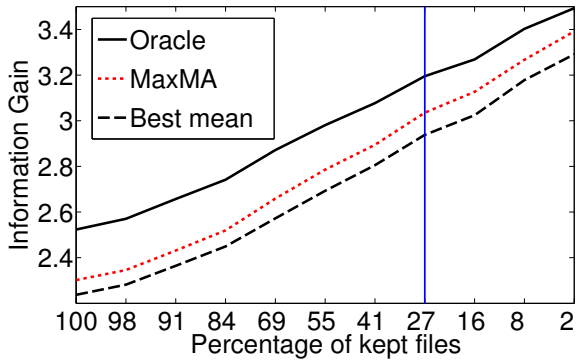
Figure 3.6 shows that applying the *MaxMA* method to choose a beat tracker leads to significant improvements when evaluating against ground truth for both Information Gain and AMLt for a wide range of thresholds. T-tests with a level of significance of $\alpha = 0.05$ were performed to compare the *MaxMA* method with the *Best mean* method at each threshold, and all differences on the left of the blue vertical lines in Figures 3.6a and 3.6b are significant. This improvement in performance occurs even when no samples are discarded and remains when retaining up to 41% for of samples AMLt and 27% for Information Gain. Beyond this point only the samples with high mutual agreement remain, which are among the easiest in the dataset, hence the choosing *MaxMA* over the *Best mean* may offer less improvement. Indeed both the MaxMa and *Best Mean* performance approach the Oracle results when only very few (easy) samples remain.

While there is still a consistent improvement for the F-measure (Figure 3.6c), this improvement is not significant for any threshold value. This is likely the result of the discontinuity of the F-measure, which assigns 0% to beat sequences misaligned in phase and values of 66% for tempo halving/doubling. These properties of the F-measure increase its variance even for sets of beat sequences that can be acceptable in terms of perceptual criteria. This supports the observation that significant differences in beat tracking performance can vary dependent on the evaluation measure (Davies et al., 2009a).

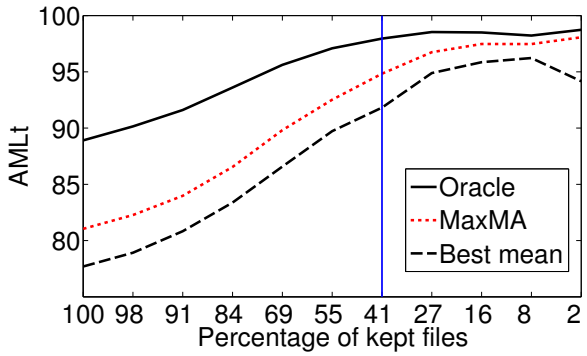
On the basis of these results, we infer that mutual agreement can be successfully applied both for choosing "beat-trackable" files and for improving beat tracking performance on these files by selecting the beat tracker that has the maximum mutual agreement with the other beat trackers. Since all beat sequences must be estimated for the file selection/rejection process, the improvement given by the *MaxMA* beat tracker choice adds negligible additional complexity.

We present in Figure 3.7 the same results of Figure 3.6b adding the performance results of the beat tracking which agrees the least (MinMA) and the *MMA* values related with the plot. These *MMA* values act as a threshold for the selection of excerpts from the dataset (*e.g.*, for an *MMA* of 2.1 bits we retain 52.1% of the song in the dataset).

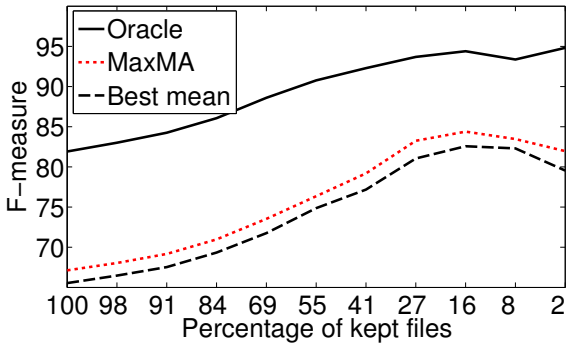
Across all *MMA* thresholds we can observe that the performance of MinMA is significantly lower than all other configurations tested. Although as seen before, lower than the Oracle, *MaxMA* outperforms the BestMean algorithm, and the difference between the two, around 3.3%, is statistically significantly ($p<0.01$) for all songs with an *MMA* below 2.4 bits. Above 2.4 bits this difference is no longer significant however the performance of the Oracle, BestMean and *MaxMA* are all very high. This suggests that for very high *MMA* thresholds, where beat tracker outputs are highly consistent with one another, any attempt to choose between the members of the committee offers little scope for improvement.



(a) Information Gain



(b) AMLt



(c) F-measure

Figure 3.6: Result of automatic beat tracker selection (*MaxMA*), compared with single best beat tracker choice (Best Mean) and oracle scores (Oracle) on Dataset1360. For the thresholds from 0 to 3 bits on the BT-MMA$_D$ the percentage of files which are kept for evaluation is depicted on the x-axis (total number of files: 1360).

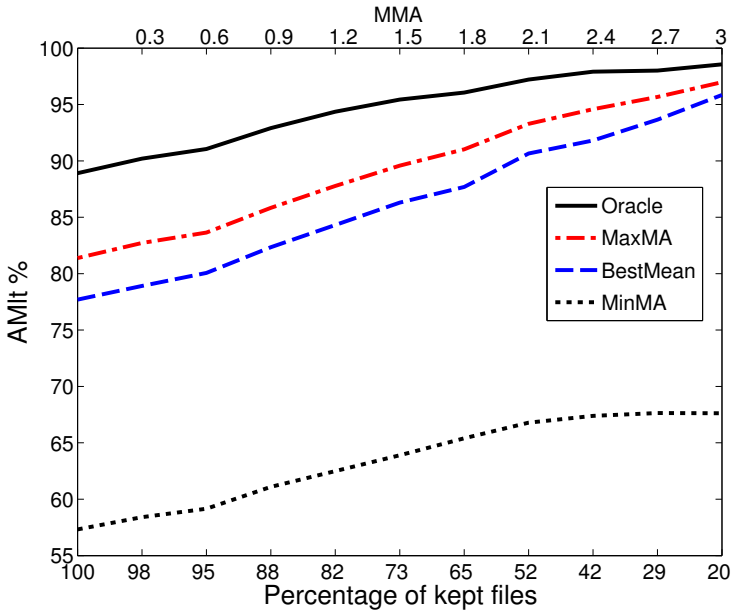


Figure 3.7: AMLt scores of the beat tracker output with maximum (*MaxMA*) and minimum (MinMA) agreement per song, compared with the single best beat tracker choice (BestMean), and the oracle score (Oracle) for various thresholds of *MMA* applied to Dataset1360.

3.3. Multi Feature beat tracking

In section 3.2.4 through the calculation of both Maximum mutual agreement (*MaxMA*), we presented a method to automatically annotate the beats in a way that exceeds the performance of the state of the art. By this method, instead of using only one beat tracker, it selects the beat estimation that most agrees (*MaxMA*) to a given piece, between beat estimations of a set of state of the art beat tracking approaches based on the *Query by committee* concept. Moreover, using *MaxMA* leads to improve the performance over consistently picking any individual algorithm from the committee of beat trackers.
Despite the good performance results, this approach is problematic because of the difference in implementation and system requirements of each algorithm. In addition, previous research in improving beat tracking presented in (Stark, 2011, chap. 4), describes the results of a modular evaluation of five state of the art beat tracking systems by comparing and contrasting their different input features and tracking models. They conclude that significant improvement in performance in beat tracking is possible by using existing tracking models

This section is based upon work in collaboration with Matthew E. P. Davies and its published in *Multi-feature beat tracking*, IEEE Trans. on Audio, Speech, and Language Processing by José R. Zapata, Matthew E.P. Davies and Emilia Gómez.

and improving the input feature. Furthermore the approach of using a single input feature for all signal types does not adequately take account of the differences between signals from different genres. and shows that by choosing a more appropriate input feature for a given signal we can achieve considerable improvements in performance.
Based on these research results, the main goal of this chapter is to extend the method of the beat trackers committee in an implementable beat tracker system that uses the query by committee idea, using a committee composed by multiple onset detection functions as inputs to one beat tracker model, and the final output is selected from the beat estimations of the committee that more agrees with the other ones.
The Figure 3.8 shows an overview of the proposed beat tracking system.

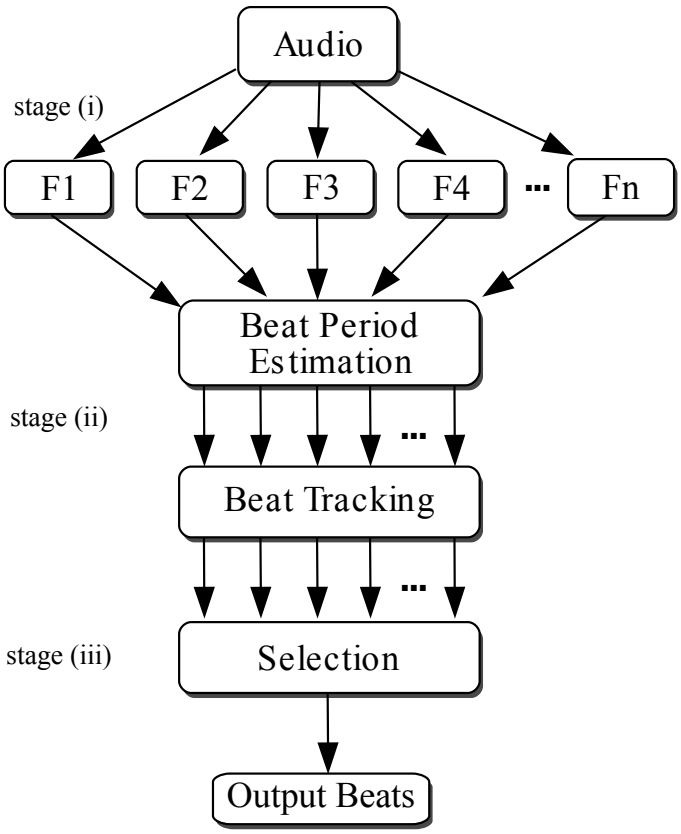


Figure 3.8: System Overview. The multi-feature beat tracker is comprised of three stages: i) a set of onset detection functions, $F1 \ldots Fn$, as input features; ii) beat period estimation and beat tracking; and iii) a selection method to choose between the set of beat outputs.

3.3.1. Proposed model

The proposed multi feature beat tracking system (shown in Figure 3.8) is composed of three parts, first, a set of onset detection functions (ODF), this is followed by beat period estimation and beat tracking for each ODF. Finally, the overall beat output is chosen using a selection method applied to the set of estimated beat locations. The proposed beat tracker and its code is publicly available[3].

3.3.2. Feature extraction

In beat tracking, an onset detection function is commonly used as a mid-level representation that reveals the location of transients in the original audio signal. This detection function is designed to show local maxima at likely event locations (Bello et al., 2005). Many methods exist to emphasize the onset of musical events and the performance of beat trackers strongly depends on the low-level signal features used at this stage (Gouyon et al., 2007).
To building our multi-feature beat tracking system we first collected the onset detection functions from each beat tracking algorithms used in Section 3.2.2, for the beat tracker committee. Some of these algorithms were freely available online and the remainder were provided by the algorithm authors or reimplemented. In addition, other onset detection functions were included, where they were deemed to be complementary to those already selected. As in chapter 3 our goal is to obtain a small but diverse committee, where each implementation is publicly available and not too demanding in terms of execution time. In addition, while a computationally efficient system is not the specific goal of this research, we seek to avoid any input features which are very computationally expensive to calculate - as their eventual benefit may not be worth the increase in computation time.
In total we compiled an initial set of nine onset detection functions which are described below. Note that while each onset detection function is extracted according to its original parametrizations in terms of window length and hop size (assuming a mono input audio signal sampled at 44.1kHz), each onset detection function is subsequently resampled to have a temporal resolution of 11.6ms prior to extracting the beats in order to match the input feature resolution expected by the beat tracking model. In the following equations for generating onset detection functions, $X(k)$ refers to the discrete Fourier transform spectrum of an audio frame x_n, the symbol k is the index over linear frequency bins in X and b is an index over a smaller number of sub-bands, B.

[3] `http://essentia.upf.edu/` , *BeatTrackerMultiFeature()*, Affero-GPL

Energy Flux (*EF*)

Equation 3.5 is a simplified implementation of the Energy flux function by Laroche (2003), and is calculated by computing short time Fourier transform frames using a window size of 2048 and hop size of 512, corresponding to an input feature resolution of 11.6ms. From these frames, each input feature sample $EF(n)$ is calculated as the magnitude of the differences of the root mean square (RMS) value between the current short time Fourier transform frame and its predecessor:

$$EF(n) = |RMS(X_n(k)) - RMS(X_{n-1}(k))| \tag{3.5}$$

Spectral Flux (*SFX*)

The spectral flux onset detection function proposed by Masri (1996) and presented in Equation 3.6, is calculated by computing short time Fourier transform (STFT) frames using a window size of 2048 and hop size of 512, corresponding to an input feature resolution of 11.6ms. From these frames, each input feature sample $SFX(n)$ is calculated as the sum of the positive differences in magnitude between each frequency bin of the current short time Fourier transform frame and its predecessor:

$$SFX(n) = \sum_{k=1}^{K} H(|X_n(k)| - |X_{n-1}(k)|) \tag{3.6}$$

where $H(x) = \frac{x-|x|}{2}$ is the half-wave rectifier function.

Spectral Flux Log Filtered (*SFLF*)

Introduced by Böck et al. (2012) this method is based on spectral flux, but the linear magnitude spectrogram is filtered with a pseudo Constant-Q filter bank, as can be seen in Equation 3.7,

$$X_n^{logfilt}(b) = log(\lambda \cdot (|X_n(k)| \cdot F(k,b)) + 1) \tag{3.7}$$

where the frequencies are aligned according to the frequencies of the semitones of the western music scale over the frequency range from 27.5 Hz to 16 kHz, using a fixed window length for the STFT, a window size of 2048 and a hop size of 512. The resulting filter bank, $F(k,b)$, has B = 82 frequency bins with b denoting the bin number of the filter and k the bin number of the linear spectrogram. The filters have not been normalized, resulting in an emphasis of the higher frequencies, similar to the high frequency content (*HFC*) method. From these frames, in Equation 3.8 each input feature sample is calculated as the sum of the positive differences in logarithmic magnitude (using λ as a

compression parameter, $\lambda = 20$) between each frequency bin of the current STFT frame and its predecessor:

$$SFLF(n) = \sum_{b=1}^{B=82} H\left(\left|X_n^{logfilt}(b)\right| - \left|X_{n-1}^{logfilt}(b)\right|\right) \tag{3.8}$$

Complex Spectral Difference (*CSD*)

The complex spectral difference input feature by Duxbury et al. (2003), presented in Equation 3.9, is calculated from the short time Fourier transform of 1024 sample frames with a 512 sample hop size, the output is interpolated resulting in a resolution of 11.6ms. The feature produces a large value if there is a significant change in magnitude or deviation from expected phase values, different from the spectral flux that only computes magnitude changes in frequency. $\tilde{X}_n$ is the expected target amplitude and phase for the current frame and is estimated based on the values of the two previous frames assuming constant amplitude and rate of phase change,

$$CSD(n) = \sum_{k=1}^{K} |X_n(k) - \tilde{X}_n(k)| \tag{3.9}$$

Beat Emphasis Function (*BEF*)

Introduced in Davies et al. (2009b), the Beat emphasis function is defined as a weighted combination of sub-band complex spectral difference functions Equation 3.9, $S_b(n)$, which emphasize periodic structure of the signal by deriving a weighted linear combination of 20 sub-band onset detection functions driven a measure of sub-band beat strength,

$$BEF(n) = \sum_{b=1}^{B=20} w(b) \cdot S_b(n) \tag{3.10}$$

where the weighting function $w(b)$ favours sub-bands with prominent periodic structure. In Equation 3.10, BEF is calculated from the short time Fourier transform of 2048 sample frames with a 1024 sample hop size, the output is interpolated by a factor of two, resulting in a resolution of 11.6ms.

Harmonic Feature (*HF*)

The harmonic feature presented by Hainsworth & Macleod (2003) is a harmonic change detection and is calculated in Equation 3.11 by computing a short time Fourier transform using a window size of 2048 sample frames with a 512 sample hop size. *HF* uses a modified Kullback-Leibler distance measure to

detect spectral changes between frequency ranges of consecutive frames. The modified measure is thus tailored to accentuate positive energy change,

$$HF(n) = \sum_{b=1}^{B} log_2 \left(\frac{|X_n(b)|}{|X_{n-1}(b)|} \right) \tag{3.11}$$

Only the region of 40Hz-5kHz was considered to pick peaks, a local average of the function was formed and then the maximum picked between each of the crossings of the actual function and the average.

Mel Auditory Feature (*MAF*)

The Mel Auditory Feature, introduce by Ellis (2007), is calculated by resampling the audio signal to 8kHz and calculating a short time Fourier transform magnitude spectrogram with a 32ms window and 4ms hop size. In Equation 3.12 each frame is then converted to an approximate "auditory" representation in 40 bands on the Mel frequency scale and converted to dB, $X^{mel}(b)$. Then the first order difference in time is taken and the result is half-wave rectified. The result is summed across frequency bands before some smoothing is performed to create the final feature,

$$MAF(n) = \sum_{b=1}^{B=40} H \left(\left| X_n^{mel}(b) \right| - \left| X_{n-1}^{mel}(b) \right| \right) \tag{3.12}$$

Phase Slope Function (*PSF*)

The group delay is used to determine instants of significant excitation in audio signals and is computed as the derivative of phase over frequency $\tau(k)$, as can be seen in Equation 3.13, in (Holzapfel & Stylianou, 2008) is used as an onset detection function: using a large overlap, an analysis window is shifted over the signal and for each window position the average group delay is computed. The obtained sequence of average group delays is referred to as phase slope function (PSF). The resulting resolution of the signal is 6.2 ms. To avoid the problems of unwrapping the phase spectrum of the signal for the computation of group delay can be computed as:

$$\tau(k) = \frac{X_{\mathbb{R}e}(k) \cdot Y_{\mathbb{R}e}(k) + X_{\mathbb{I}m}(k) \cdot Y_{\mathbb{I}m}(k)}{|X(k)|^2} \tag{3.13}$$

Where $X(k)$ and $Y(k)$ are the Fourier Transforms of $x[n]$ and $nx[n]$, respectively. The phase slope function is then computed as the negative of the average of the group delay function.

Bandwise Accent Signals (*BAS*)

Introduced by Klapuri et al. (2006), Bandwise Accent Signals are calculated from 1024 sample frames with a 512 sample hop size. The Fourier transform of these frames is taken and used to calculate power envelopes at 36 sub-bands on a critical- band scale. Each sub-band is up-sampled by a factor of two, smoothed using a low-pass filter with a 10-Hz cutoff frequency and half-wave rectified. A weighted average of each band and its first order differential is taken, $u_b(n)$. Finally, in Equation 3.14 each group of 9 adjacent bands (i.e. bands 1–9, 10–18, 19–27 and 28–36) are summed up to create a four channel (c) input feature with a resolution of 5.8 ms. We sum these four channels to generate a single output feature,

$$BAS(n) = \sum_{c=1}^{4} \sum_{b=9(c-1)+1}^{9c} u_b(n) \qquad c = 1, \ldots, 4 \tag{3.14}$$

3.3.3. Beat period estimation and tracking model

Given each onset detection function we now address the task of estimating beat locations. Since our system relies on a single beat tracking model, we select a beat tracker which has been shown to perform well in comparative studies and is freely available[4]. To this end, we choose the method of Degara et al. (2012), which was also part of the committee of beat tracking algorithms in section 3.2.2.

The core of Degara's beat tracking model is a probabilistic framework which takes as input an onset detection function (used to determine the phase of the beat locations) and a periodicity path which indicates the predominant beat period (or tempo) through time. While a different input feature (or user-specified input) could be used to determine the periodicity path, in practice it is estimated from the same onset detection function. The technique for finding the periodicity (as used in (Degara et al., 2012)) is an offline version of the Viterbi model in (Stark et al., 2009). This Viterbi model assumes the beat period to be a slowly varying process with transition probabilities modeled using a Gaussian distribution of fixed standard deviation. To estimate the beats, the system integrates musical-knowledge and signal observations using a probabilistic framework to model the time between consecutive beat events and exploits both beat and non-beat signal observations. For more information on the tracking method, see (Degara et al., 2012). Since our primary concern in this paper relates to the input features supplied to the beat tracker, we can treat the beat tracker as a "black box". To create the committee of beat trackers, we calculate a separate periodicity path and set of beat locations for each onset detection function.

[4] http://www.gts.tsc.uvigo.es/~ndegara/Publications.html

3.3.4. Selection method and measuring mutual agreement

In Section 3.2.4 we use the Maximum Mutual Agreement *MaxMA* as a selection method to obtain an output between a set of beat tracker beat estimations, this selection leads to significant improvements when evaluated against the ground truth and compared with a set of 16 state of the art beat tracking systems.
In Section 3.1.1 we reviewed the properties of existing evaluation measures and selected the Information Gain (InfGain) approach by Davies et al. (2011) as the only measure with a true zero value, able to match low *MMA* (measured in bits) with unrelated beat sequences.
While Information Gain was shown to be a good indicator of agreement between beat sequences from among existing beat tracking evaluation methods, it is not the only approach which could be used. In this section we also explore an alternative mechanism for measuring agreement, the *regularity function* of Marchini & Purwins (2011), which quantifies the degree of temporal regularity between time events.
Regularity: Quantifies the degree of temporal regularity between beat estimations. Firstly, to calculate the regularity we first concatenate and sort the beats of two different beat sequences, then we compute the histogram of the time differences between all possible combinations of two beats (the complete inter-beat interval histogram, *CIBIH*). In this way, we obtain is a kind of "harmonic series" of peaks that are more or less prominent according to the self-similarity of the sequence at different time scales. Second, we compute the autocorrelation $ac(t)$(where t corresponds to lag in seconds) of the *CIBIH* which, in the case of a regular sequence, has peaks at multiples of the tempo. Let t_{usp} be the positive time value corresponding to its upper side peak. Given the sequence of m beats $x = (x_1, ..., x_m)$ we define the regularity of the sequence of beats x to be:

$$\textit{Regularity(x)} = \frac{ac(t_{usp})}{\frac{1}{t_{usp}} \int_0^{t_{usp}} ac(t_{usp})}. \tag{3.15}$$

If the beat estimations are more equally spaced in time the regularity value will be higher, whereas if the beat estimations are unrelated the regularity value will be lower. For more Information see (Marchini & Purwins, 2011).
Referring again to Figure 3.8, the chosen selection mechanism (either Information Gain or Regularity) is the final stage in our multi-feature beat tracking which provides the eventual beat output.

3.4. Experimental setup

3.4.1. Dataset

The Evaluation was realized using **Dataset1360**, the largest available dataset for beat tracking evaluation that contains a total of 1360 excerpts presented

in Section 3.2. We use Dataset1360 to analyze the diversity and accuracy of the onset detection functions. Based on these results we will i) select our committee of onset detection functions ii) give proof of using Maximum Mutual Agreement (*MaxMA*) for selecting the best beat tracking estimation from the committee and iii) verify the behavior of the *MMA* method calculated with a committee formed by different onset detection functions to assess difficulty for automatic beat tracking.

3.4.2. Evaluation measures

For evaluating the beat tracking accuracy against manual annotations, we use a subset of methods from the beat tracking evaluation toolbox[5] (Davies et al., 2009a). These evaluation methods are also used in the beat tracking evaluation task within MIREX.
Among all the proposed evaluation metrics, we use the continuity measures as originally defined in (Hainsworth, 2004; Klapuri et al., 2006) with an output range between [0 - 100]%. This allows us to analyze both the ambiguity associated with the annotated metrical level and the continuity in the beat estimates. These accuracy measures consider regions of continuously correct beat estimates relative to the length of the audio signal analyzed. Continuity is enforced by defining a tolerance window of 17.5% relative to the current inter-annotation-interval. To allow the beat tracker to initially induce the beat, events within the first five seconds of each excerpt are not evaluated. The continuity-based criteria used for performance evaluation are the following:

- **CMLc** (Correct Metrical Level with continuity required) gives information about the longest segment of continuously correct beat tracking.
- **CMLt** (Correct Metrical Level with no continuity required) accounts for the total number of correct beats at the correct metrical level.
- **AMLc** (Allowed Metrical Level with continuity required) is the same as **CMLc** but it accounts for ambiguity in the metrical level by allowing for the beats to be tapped at double or half the annotated metrical level.
- **AMLt** (Allowed Metrical Level with no continuity required) is the same as **CMLt** but it accounts for ambiguity in the metrical level.

3.4.3. Reference systems

To compare our system against existing beat trackers, we compiled a set of existing algorithms, including those with freely available implementations on-line and others provided by the authors of the systems on request. To summarize the accuracy of each beat tracking system, the mean value of the performance

[5] http://code.soundsoftware.ac.uk/projects/beat-evaluation

measures across all the audio files of the test database is presented. Statistically significant difference on the mean values are also checked. For this, we use a paired T-test with $\alpha = 0.05$ as a guide to indicate statistical significance. In total we compiled 20 state of the art beat trackers, expanding the set originally in Table 3.1 with the approaches by Krebs & Widmer (2012), Gkiokas et al. (2012), Khadkevich et al. (2012) and a commercial approach by the Echonest[6], and also compare against the five committee beat tracker (Section 3.2.2) .

3.5. Results

3.5.1. Committee members

Before presenting comparative results against other beat tracking algorithms we first analyze the composition of the committee of beat trackers in our multi-feature beat tracker. The committee is composed of the beat tracking estimation from the following onset detection functions: bandwise accent signal (*BAS*), beat emphasis function (*BEF*), complex spectral difference (*CSD*), energy flux (*EF*), harmonic feature (*HF*), mel auditory feature (*MAF*), phase slope function (*PSF*), spectral flux (*SFX*) and spectral flux log filtered (*SFLF*). To find the relevance of each ODF in the committee we make use of the sequential forward selection, SFS, method, as used in Section 3.1.2. We ran the Degara beat tracker with each onset detection function on *Dataset1360* and measure the per track performance of the beats resulting from each ODF. As the first member of the committee we select the ODF with the best mean performance across the entire dataset. Then we iteratively add a new member to the committee based on whichever ODF (combined with those already in the committee) leads to the best *Oracle* performance - i.e. by choosing a priori the best beat sequence per excerpt in the dataset. This procedure is iteratively continued until all onset detection functions have been included.
Once this selection process has been completed we can look at the order in which each ODF entered the committee and the improvement in performance achieved by its inclusion. The mean performance of each feature in the *Dataset1360* is presented in Table 3.4, from which we can see the *CSD* performs best, and the *EF* and *PSF* have lowest overall accuracy, perhaps due to the specific emphasis in detecting changes in only one signal variable (i.e. energy, or phase), compared to the more general nature of the *CSD* method.
We can determine a subset by fixing the number of committee members at the point where improvements offered by additional members is small. A choice of beat trackers guided by this strategy takes into account both accuracy and diversity.

[6] http://developer.echonest.com/

Table 3.4: Mean Continuity measures performance (%) of each feature and the Oracle in the 1360 Song Dataset, sort by sequential forward selection method

ODF	CMLc	CMLt	AMLc	AMLt
CSD	**46.1**	**50.3**	**69.8**	**77.6**
HF	38.5	45.6	62.0	73.7
PSF	31.1	35.2	61.3	69.9
EF	39.6	44.7	56.1	64.6
SFLF	44.2	48.0	68.9	76.8
BEF	38.0	42.2	65.5	73.5
BAS	43.0	46.8	68.5	76.4
MAF	42.2	46.8	63.9	73.0
SFX	43.2	47.9	65.8	73.9
Oracle	64.9	69.0	85.5	90.5

Using the SFS method the order in which the ODF enter the committee is as follows: Complex spectral difference (*CSD*), Harmonic function (*HF*), Energy Flux (*EF*), Phase slope function (*PSF*), Spectral flux logarithmic filtered (*SFLF*), Beat emphasis function (*BEF*), Bandwise accent signal (*BAS*), Mel auditory function (*MAF*) and spectral flux (*SFX*).
In Figure 3.9(a) a comparison between the mean performance of the Oracle and the Multi-feature beat tracker versus the number of committee members is presented. By comparing the improvements between the best ODF alone (*CSD*) when new members (given by the SFS method) are added to the committee, we find that after the sixth member is added, the performance is higher and statistically significant for the **AMLc** and **AMLt** measures. In Figure 3.9(b) we show the improvement obtained by automatic selection between beat outputs using information gain and regularity. Table 3.5 presents the evaluation results of the best ODF mean performances of each of the genres of *Dataset1360* per evaluation measure. There is no statistical difference between the results of the best three ODFs per genre. However some ODFs performed statistically worse than the others in these genres: Acoustic (*EF*), Afro-American (*PSF*), Classical (*BEF*, *EF*, *SFX*), Classical Solo (*SFX*), Electronic (*PSF*), Jazz (*EF*, *HF*, *MAF*), Rock/Pop (*PSF*), Samba (*HF*, *MAF*). Overall our results confirm the intuition that ODFs which are sensitive only to phase or harmonic changes are not the best choice for music genres with strong percussion, furthermore the *EF* is not a good choice for music without prominent percussion. Comparing the **AMLt** results of each onset detection function in *Dataset1360*, we find that 53% of the songs could be improved by using multiple ODF versus only using the single best performing onset detection function for this model, which led to an 11.6% average improvement.

Table 3.5: Mean performance (%) of the best feature per genre in the 1360 Song Dataset

Genre	CMLc	CMLt	AMLc	AMLt
Acoustic	39.8 (*SFLF*)	45.6 (*SFLF*)	57.1 (*SFLF*)	67.6 (*SFLF*)
Afro-American	70.8 (*SFX*)	73.4 (*SFX*)	85.6 (*CSD*)	93.3 (*CSD*)
Balkan	19.6 (*EF*)	20.9 (*EF*)	77.4 (*SFLF*)	83.0 (*SFLF*)
Choral	8.8 (*PSF*)	13.9 (*PSF*)	16.4 (*HF*)	32.2 (*HF*)
Classical	38.7 (*BAS*)	47.0 (*BAS*)	53.9 (*BAS*)	67.5 (*HF*)
Classical Solo	31.0 (*BAS*)	33.3 (*BAS*)	66.1 (*BAS*)	73.6 (*BAS*)
Electronic	55.8 (*CSD*)	58.7 (*EF*)	81.6 (*SFX*)	83.6 (*SFX*)
Jazz	48.5 (*SFLF*)	54.9 (*CSD*)	68.1 (*SFLF*)	78.4 (*CSD*)
Rock/Pop	62.5 (*CSD*)	65.8 (*CSD*)	82.6 (*CSD*)	88.9 (*CSD*)
Samba	52.2 (*CSD*)	53.1 (*CSD*)	67.0 (*BEF*)	68.5 (*BEF*)

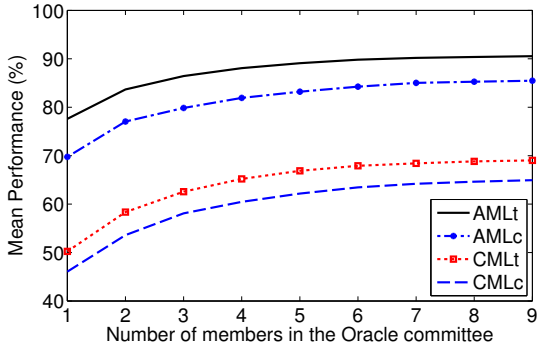
3.5.2. Comparison results

In Table 3.6, the mean accuracy of the different beat tracking algorithms is compared. We present two configurations of the multi feature beat tracker: the first one with only six committee members (*CSD*, *HF*, *EF*, *PSF*, *SFLF* and *BEF*) chosen by the SFS method, for this configuration, the information gain (MultiFt InfG) and regularity (MultiFt Reg) were used in the selection step of the proposed algorithm; and the second configuration (MultiFt Essentia) which is the C++ of the Multi-feature Information Gain (ZDG1) (Zapata et al., 2012a) submitted to the MIREX 2012 beat tracking task using *CSD*, *HF*, *EF*, *BEF* and *MAF*, this configuration is the released version of the Multi-feature beat tracker due to the computational cost of including the *PSF* ODF.

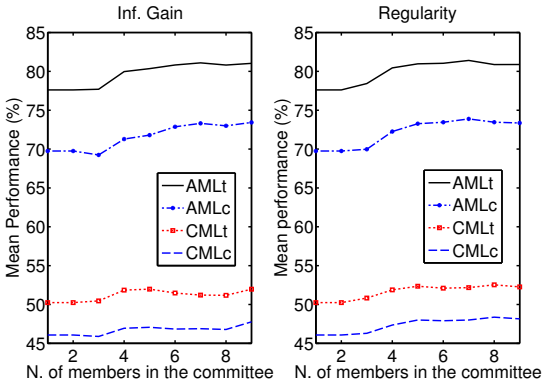
While the mean performance of all beat tracking systems is moderately low when using **CMLc** or **CMLt** (i.e., when the beats must be tapped at the annotated metrical level), performance naturally improves when we incorporate the additional, "allowed" metrical levels using **AMLc** and **AMLt**.

When comparing the proposed beat tracking algorithm with the reference systems, as shown in Table 3.6, we see that the proposed method outperforms the reference methods in the mean value for all of the evaluation criteria. However, not all of the differences are statistically significant ($p<.05$). We find no significant differences between the proposed algorithm and the following reference methods with the DAV, DEG and KLA systems under **CMLc**, and then with the KLA system under **CMLt**.

When we compare MultiFt InfG and MultiFt Reg, which uses a subset of six ODFs, with MultiFt Essentia which uses a subset of five ODFs, we do not find statistically significant differences. Furthermore we do not find any statistical difference compared to the committee system with five separate beat tracking algorithms (Beatroot, Degara, Ellis, Klapuri, IBT) as proposed in Section 3.2.2.



(a)



(b)

Figure 3.9: (a) Oracle Mean Performance vs number of committee members. (b) Multi Feature (Inf Gain and Regularity) Mean Performance vs number of committee members.

3.5.3. Automatic selection results

To verify that using either Information Gain or Regularity as a selection mechanism provides a significant improvement, we can compare the performance in Table 3.6 for our system with what happens if we make a random selection of the "best" beat tracking output per song. Running multiple trials we obtained mean performance of [40.6%, 45.3%, 64.6%, 73.3%] with variance [0.49, 0.50, 0.37, 0.32] for **CMLc**, **CMLt**, **AMLc**, **AMLt** respectively. The increase in performance we achieve using a structured, rather than random selection process is highly significant ($p<.00001$). However, the beat tracking accuracy from using either Information Gain or Regularity as a selection method falls well below the theoretical optimum of the Oracle system which can choose the best beat sequence per song, suggesting that automatic selection methods remains a profitable avenue for future work.

Table 3.6: Mean ground truth performance of each BT on *Dataset1360*. Bold numbers indicate best performances.

Beat Tracker	CMLc (%)	CMLt (%)	AMLc (%)	AMLt (%)
Aubio (Brossier, 2006)	26.43	35.12	37.73	50.57
Beat.e (Krebs & Widmer, 2012)	36.13	42.19	61.59	73.95
Beatit (Bonada & Gouyon, 2006)	6.98	8.69	43.64	60.95
Beatroot (Dixon, 2007)	29.05	35.70	53.51	70.84
BeatUJAEN (Mata-Campos et al., 2010)	10.45	17.17	26.84	41.63
Boeck (Böck & Schedl, 2011)	31.46	43.48	42.20	58.74
BpmHistogram (Aylon & Wack, 2010)	13.82	21.60	34.38	57.32
Davies (Davies & Plumbley, 2007)	46.82	50.79	69.28	75.88
Degara (Degara et al., 2012)	46.04	50.17	69.89	77.72
Echonest http://developer.echonest.com	31.68	36.32	52.01	59.83
Ellis (Ellis, 2007)	10.66	14.02	38.54	60.03
Gkiokas (Gkiokas et al., 2012)	41.47	47.10	62.73	72.75
Hainsworth (Hainsworth & Macleod, 2004)	34.28	37.24	54.08	59.62
IBT causal (Oliveira et al., 2012)	25.27	30.82	47.05	58.0
IBT non-causal (Oliveira et al., 2012)	32.54	36.88	63.97	73.76
Klapuri (Klapuri et al., 2006)	47.75	**52.71**	69.79	77.70
Lee (Lee, 2010)	1.61	7.06	5.87	26.38
Scheirer (Scheirer, 1998)	21.19	34.52	30.38	48.97
Shine (Khadkevich et al., 2012)	45.23	48.67	62.52	67.70
Stark (Stark et al., 2009)	41.68	47.32	61.64	70.99
Multi-Feature InfGain	46.8	51.5	**72.9**	80.8
Multi-Feature Regularity	**47.9**	**52.1**	**73.5**	**81.0**
Multi-Feature Essentia	46.16	50.67	71.91	80.37
5 BT Committee, (Section 3.2.2)	46.88	51.55	72.28	**81.39**
Oracle	64.95	69.02	85.47	90.54

3.5.4. MIREX results

Thus far, all of our analysis has been *Dataset1360*, and while there is a wide diversity of musical genres and a large number of annotated files, we should acknowledge that the performance we observe might be slightly optimistic given access to the test data when choosing the committee members. Therefore, in addition to our own evaluation on the *Dataset1360*, we also report results from the 2012 MIREX Audio Beat Tracking task, where we submitted two versions of our multi-feature beat tracker: ZDG1 and ZDG2 (Zapata et al., 2012a) which used *BEF*, *CSD*, *EF*, *HF* and *MAF* as committee members, and used the information gain and regularity selection methods respectively. The MIREX dataset is private and therefore can be considered as appropriate validation for our method.
In the Table 3.7 we show the 2012 MIREX results (sorted by **AMLt**) for the beat tracking task are presented, and also the best **AMLt** performers in

2011, 2010 and 2009 in the MCK dataset. The MCK dataset contains 160 30-sec. excerpts (WAV format) and has been used since the beginning of the MIREX beat tracking evaluation in 2006. These audio recordings had a stable tempo value, a wide distribution of tempi values, and a large variety of instrumentation and musical styles. About 20% of the files contain non-binary meters, and a small number of examples contain changing meters.
As can be seen from the table, our multi-feature systems ZDG1 and ZDG2 perform competitively with the submitted algorithms for 2012 and those which have performed well in previous years. While the differences in performances are small between the most accurate systems, we believe that, to date, ZDG1 has the highest reported accuracy on the MCK dataset for **AMLc** and **AMLt**.

Table 3.7: MIREX 2012 mean performance (%) and the best **AMLt** performance in 2011,2010 and 2009 in MCK dataset

Year	Beat Tracker	CMLc	CMLt	AMLc	AMLt
2012[7]	**ZDG2**	25.0	33.4	**51.8**	**66.7**
	GP3	23.7	33.7	49.3	66.5
	ZDG1	23.7	32.3	49.5	65.1
	GP2	23.3	32.3	48.6	64.9
	GKC2	**25.8**	32.9	51.0	64.2
	ODGR1	21.6	20.0	49.4	64.2
	FK1	22.3	35.1	41.5	63.3
	ODGR2	22.4	30.4	47.0	62.7
	KB1	17.5	29.9	35.9	60.2
	ODGR3	21.8	29.7	44.2	59.7
	FW4	23.7	34.5	42.4	59.1
	KFRO1	25.0	32.0	47.1	58.8
	ODGR4	20.0	28.3	41.4	58.2
	SB6	20.4	29.3	40.8	57.2
	FW3	22.5	34.1	39.2	57.0
	SB3	20.8	30.0	37.2	53.6
	GP4	19.6	30.4	35.2	52.5
	SB7	16.5	26.4	27.6	44.2
	SB4	14.2	24.0	24.4	42.1
	FW5	9.4	18.8	17.0	34.8
2011[8]	GP5	24.0	33.7	49.3	66.5
2010[9]	GP3	24.0	33.7	49.0	66.1
2009[10]	GP1	26.0	**35.5**	49.1	66.6

[3] http://nema.lis.illinois.edu/nema_out/mirex2012/results/abt/mck/
[4] http://nema.lis.illinois.edu/nema_out/mirex2011/results/abt/mck/
[5] http://nema.lis.illinois.edu/nema_out/mirex2010/results/abt/mck/
[6] http:/music-ir.org/mirex/wiki/2009:Audio_Beat_Tracking_Results

3.6. Conclusions and future Work

In this chapter, we present a technique, based on Mean Mutual Agreement (*MMA*) and the selection of the beat tracker output which most agrees with the remainder of the committee (*MaxMA*), that automatically annotate the beats in a way that exceeds the performance of the state of the art.
The fact that a simple approach of this kind was able to demonstrate a significant improvement over using individual state of the art algorithms is encouraging. Yet, as our results indicate, performance of *MaxMA* falls some way below that of the Oracle system (selecting the best performance per song) using our committee of beat trackers or the proposed model with information gain and regularity. This suggests that there is still room for making a more accurate selection among existing algorithms, and exploring new selection methods will form a further area for future work.
We demonstrated that the choice of the evaluation measure to compute the mutual agreement is crucial, and that the Information Gain was better suited to this task than both the F-measure and AMLt evaluation methods. However, the Information Gain method appears less effective in highlighting where beat tracking algorithms strongly agree with each other. Hence in future work we will explore methods to combine the information from different evaluation methods.
The proposed Multi feature beat tracker system was compared to 20 reference systems in a large beat tracking annotated dataset and outperformed all the reference systems in the mean value under all the evaluation criteria used. We found significant statistical differences in all of the measures on 17 of the 20 references systems against the Multi feature and the five beat tracker committee. Moreover the improvement in the AMLc and AMLt measures are statistically significant when compared with all the reference systems.
The Multi feature beat tracker achieved better results when uses existing onset detection functions and a tracker model, contrary to recent work in the field, than designing more complex tracking models. Because the beat tracker could be improved with other onset detection functions, we encourage the research community to work on this subject by trying other onset detection functions, mixing the existing ones (Stark, 2011, chap. 4) or enhancing the periodicity characteristics of the audio signal with other techniques like source separation or voice reduction.

3.7. Summary

In this chapter, we present a beat tracking strategy based on the Maximum mutual agreement (*MaxMA*) method to select the best beat tracking output from a committee of beat trackers. This method improves the accuracy of the beat estimations over consistently picking any individual algorithm from the committee. To build our system we select a set of 5 beat trackers over 16 evaluated beat trackers, our method, is based on the measurement of mutual agreement between beat estimations and we determine the influence of choosing the Information gain to computed the mutual agreement.
Finally, we present an stand-alone beat tracker (Multi feature Beat tracker) that extends the method of the beat trackers committee (Section 3.2.2). The Multi feature beat tracker uses a committee composed by multiple onset detection functions and one beat tracker model. The final output is selected by using the Maximum mutual agreement from the beat estimations outputs of each onset detection function. The Multi feature Beat Tracker outperforms the state of the art beat trackers on the evaluated measures, and its evaluation performance is statistically comparable to the five beat trackers committee.

CHAPTER 4

Improving Beat Tracking

Has beat tracking reached the upper limit of performance (the so-called "glass-ceiling" effect)? and no further gains in performance are possible ? Perhaps a more likely explanation for the current stagnation in performance lies in the data used to evaluate beat trackers. We believe the continual re-use of existing datasets, e.g. Dixon (2007); Hainsworth & Macleod (2004); Klapuri et al. (2006), has led to an (somewhat) inevitable over-fitting of beat tracking algorithms to the limited data which is available. Furthermore, within these existing databases, there is a bias towards musical styles whose beats can be more easily tracked; genres typically characterized by clear percussive content and steady tempi like rock, pop and electronic dance. This preference towards easier musical styles means that challenging excerpts, where beat tracking algorithms fail, are typically treated as outliers and little effort is made to determine how to process them.

Given the hypothesis that a glass ceiling in beat tracking exists due to a lack of diversity in annotated data, an appropriate strategy to address it would be to annotate more musical examples. However, the manual annotation of beat locations can be extremely difficult and time-consuming. Therefore, it makes sense to restrict annotation to music examples which are in someway informative for the beat tracking problem. To this end, our approach is to focus on the selection of musical pieces that are shown to be difficult for current state of the art systems. Since the goal is to subsequently derive ground truth annotations, this estimation of difficulty must be achieved without any ground truth annotations.

While some effort has been made to estimate rhythmic difficulty, this has typically been limited in scope by focusing on measures of beat strength (Goto, 2001; Tzanetakis et al., 2002). Furthermore, these methods have not been used for the selection of music samples to annotate. A related study of difficulty

This section is based upon work in collaboration with Andre Holzapfel, Matthew E. P. Davies, João Lobato Oliveira and Fabien Gouyon. This is a compilation of papers published in a journal and peer reviewed conference, Holzapfel et al. (2012b); Zapata et al. (2012b)

in beat tracking by Grosche et al. (2010) considered local properties of compositions that cause beat trackers to stumble. Our interest is on the global properties of musical excerpts.
In chapter 3, we present the correlation between the *mean ground truth performance* of all beat trackers (BT-MGP) and the mean mutual agreement between the beat estimation of the beat trackers (BT-MMA). Using this information we found when the *MMA* value is low the MGP is low and vice when the *MMA* value is high, the MGP value is high. To improve the beat tracking estimation, we want to take advantage of the *MMA* value to devise a method to identify the specific musical characteristics that negatively affect beat tracking performance and identify the *MMA* value where the automatic beat tracking is successful without the need of ground truth.
We used the proposed Mean Mutual Agreement (*MMA*) method to build a dataset of samples that are problematic for beat trackers. Listeners were then asked to tap the beat of those detected samples in a spontaneous manner, describe the signal properties, and eventually to determine ground truth beat annotations. This data was used to examine similarities and differences between human listeners and automatic beat tracking. Results demonstrate that among the files show to be difficult for beat trackers some were perceptually easy for human tappers, while files characterized by expressive timing and/or quiet accompaniment were considered equally difficult. We believe that the highest potential for improving beat tracking technology lies in setting the methods to address those files whose cause failure of beat trackers. However, we focus on files that contain a perceivable beats rather than attempting to address those which human tappers also struggled with.
The remainder of this chapter is structured as follows; in Section 4.1, we use the mutual agreement to build a new dataset with problematic cases for beat tracking and we we provide details about a new dataset and the annotation process. In Section 4.1.2, we research the difficulty of the new dataset for both, automatic beat tracking and human listeners. Based on the analysis of the difficulties in beat tracking, in Section 4.2, we evaluate and discuss how voice suppression techniques improve rhythmic saliency in songs with highly predominant vocals and quiet accompaniment. In Section 4.3, we describe the application of our *MMA* method in order to identify and reject musical pieces where beat tracking will fail, and we demonstrate how beat tracking performance can be improved by inspecting the properties of the beat tracking committee applying the technique to non-annotated data. Moreover, we define a *MMA* perceptual confidence threshold (Section 4.3.1) to determine a "sucess" in the automatic beat tracking process. Additionally, those songs with *MMA*, above the perceptual confidence threshold (Section 4.3.2), were automatically beat annotated in a way that exceeds the performance of the state of the art. Finally, in Section 4.5 we conclude with a discussion of the results and areas for future work.

4.1. Building a challenging dataset

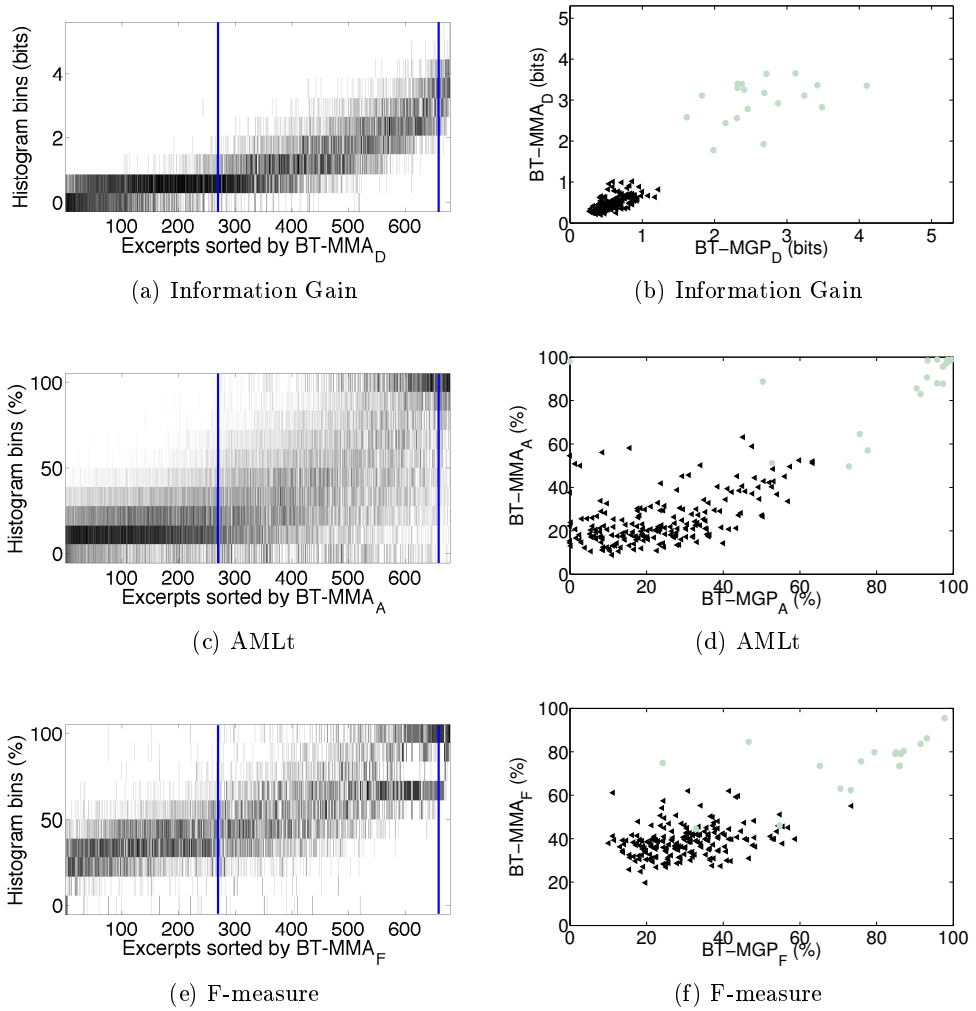
We start from the assumption that adding diversity to existing collections is necessary to facilitate future improvement in beat tracking systems. To this end we now describe a new dataset and compare its properties to those of **Dataset1360**. The new dataset was compiled by choosing a set of CDs and extracting 40s of each song. We chose music that could be considered difficult in terms of their rhythmic properties. For this, we concentrated on styles of Western music, because it is not apparent how the notion of beat could be applied to music of other cultures. The CDs contained a variety of styles including, classical music, romantic music, film soundtracks, blues, chanson, and solo guitar compositions. We extracted a total of 678 excerpts.
A subset of the 678 pieces was chosen for manual annotation with the goal of selecting those pieces that cause the largest problems to the beat tracking approaches. We decided to choose samples with BT-MMA$_D$ values $\leq$ 1 bit, this resulted in 270 samples. Because for values $\leq$ 1 bit, the histograms in Figure 3.4a have a clear peak and the correlation with BT-MGP$_D$ in Figure 3.4b is strong. We do not intend for this threshold to be interpreted as a globally valid division between easy and difficult files, rather it was chosen empirically to maximize the probability of obtaining only difficult files. In order to cross-check the assumption of those files being difficult, we added 19 samples with the highest BT-MMA$_D$ value which should be characterized by a high BT-MGP. This set of 289 pieces that are chosen for annotation will be referred to as **DatasetSMC** throughout the remainder of the work.
The first step consists of recording *spontaneous taps* from all authors of this work for all 289 pieces in DatasetSMC. The taps enable us to examine the ability of listeners to follow the beat in a possibly difficult piece of music without any entrainment. The *MMA* of these tappings is used to assess the perceptual difficulty, and will be compared with the *MMA* of automatic beat trackers. It should be stated that all authors come from an engineering background, but four had many years of experience as practicing musicians in different styles and instruments. Before tapping, each subject was not allowed to listen to the piece. They were instead asked to tap the beat of the piece while listening to it for the first time and without the possibility of correcting the taps afterwards.
In the next step the files in DatasetSMC were equally distributed among the authors for ground truth annotation. The annotations were performed using Sonic Visualiser (Cannam et al., 2010). To assist with the annotation, each annotator was allowed to use multiple visualizations such as the waveform or spectrogram, but the use of automatic beat tracking or onset detection algorithms was not permitted, however, spontaneous taps could be used. When available, scores of the pieces were used as a guideline to arrive at a valid annotation especially for classical and romantic music. Each annotator was given the option to reject a file if the annotation process appeared intractable. This happened in 72 cases, resulting in 217 valid beat annotations for DatasetSMC.

Finally, the annotator had to compile a tag file for each annotated sample. The tags specify which signal characteristics either made the annotation difficult or might caused errors in automatic beat tracking algorithms. An arbitrary number of tags could be assigned to a song, if none of the tags applied to a song the tag "none" was used. The list of tags and their rate of appearance in DatasetSMC will be presented in Section 4.1.2.
Each annotation was subsequently evaluated by a second subject. In the annotation process all annotators expressed insecurity about some of their annotations due to the high rhythmic complexity of some of the files. In order to cope with this problem we decided to consult experts[1] with conservatory degrees in music and composition, and with their assistance we were able to obtain a more reliable ground truth especially for the most difficult samples.

4.1.1. Automatic beat tracking on the new dataset

For DatasetSMC, BT-MMA histograms and scatter plots of BT-MMA over BT-MGP are depicted in Figure 4.1. Computations were performed in the same way as for **Dataset1360**, enabling for a comparison between Figure 3.4 and Figure 4.1. A common characteristic of the plots for Dataset1360 and DatasetSMC is the high correlation between BT-MGP and BT-MMA for small values when using the information gain measure (see Figures 3.4b and 4.1b), respectively. Again, for F-measure and AMLt such a correlation cannot be observed. This provides strong evidence for using BT-MMA$_D$ to detect difficult files in the context of the newly annotated DatasetSMC.
Differences between **Dataset1360** (Table 3.3) and **DatasetSMC** (Table 4.1) are evident for all three evaluation measures: The mutual agreement histograms in the left columns are strongly biased towards the upper right corner for Dataset1360 and towards the lower left corner for DatasetSMC. Again, the histograms for BT-MMA$_D$ in Figure 4.1a show a more accentuated concentration and a continuous development from concentration in low to concentration in high histogram bins. However, in Figure 4.1a a higher proportion of histograms is characterized by a concentration in bins of 1 bit or less. This indicates that DatasetSMC contains a larger relative percentage of difficult samples than Dataset1360. The super-imposed blue vertical lines in the histogram plots in Figure 4.1 indicate the borders for the initial choice of files to be annotated, i.e., the first 270 files and the last 19 files sorted by BT-MMA$_D$ (see Section 4.1). Samples on the left of the first line were chosen because they were assumed to be difficult (low BT-MMA$_D$), while the 19 files on the right of the second line in the histogram plots have been included because they were supposed to be the easiest in the dataset (high BT-MMA$_D$). In Figure 4.1b a clear separation can be observed between those files, where the difficult files are marked by black triangles and the easy files by gray circles. This separation

[1] We thank Michael Hecht and the group at Butler school of music in UT Austin, for assistance in improving the annotations



(a) Information Gain (b) Information Gain

(c) AMLt (d) AMLt

(e) F-measure (f) F-measure

Figure 4.1: Left side: Each column depicts the histogram obtained from the $5 * 4/2$ mutual agreements of the beat sequences for a song in the 678 samples that were used to derive DatasetSMC, histograms are sorted by their mean values (BT-MMA). Dark colors indicate high histogram values. Files that were excluded from annotation lie between the blue lines. Right side: *MMA* versus MGP scatter plots for the annotated 217 files in DatasetSMC. Pieces which are supposed to be easy according to their BT-MMA are depicted by gray circles.

is not evident for the other evaluation measures in Figures 4.1d and 4.1f, and the difficult files form wider spread clusters.

The D-MGP values for DatasetSMC are depicted in Table 4.1, bold numbers indicate the best beat tracking results without statistically significant differences. Note that the files in DatasetSMC were selected based on BT-MMA$_D$ and are supposed to be difficult, with the exception of the included 19 files

Table 4.1: Mean ground truth performance of each BT (D-MGP) on **DatasetSMC**. Bold numbers indicate best performances.

Beat Tracker	AMLt (%)	F-measure (%)	Inf. Gain (bits)
Aubio (AUB)	18.5	24.7	0.68
Beatit (BIT)	20.6	28.7	0.53
Beatroot (DIX)	**27.6**	32.2	0.66
BeatUJAEN (BUJ)	23.9	27.7	0.60
Böck (BOE)	**26.1**	**40.1**	**0.91**
Davies (DAV)	**33.4**	32.2	**0.90**
Degara (DEG)	**33.4**	**34.6**	**0.89**
Ellis (ELL)	20.8	**35.2**	0.62
BpmHistogram (BHI)	23.3	26.6	0.64
Hainsworth (HAI)	**26.0**	24.8	**0.83**
IBT causal (IB1)	21.1	26.8	**0.70**
IBT non-causal (IB2)	**28.6**	31.1	**0.78**
Klapuri (KLA)	**33.9**	**36.2**	**0.92**
Lee (LEE)	12.9	**34.6**	0.50
Scheirer (SCH)	18.5	30.2	**0.70**
Stark (STA)	**26.0**	27.3	**0.74**
Mean	22.7	30.8	0.73
Random	18.0	25.0	0

with high BT-MMA$_D$. Indeed, for DatasetSMC the overall performance is much lower than for Dataset1360 (see Table 3.3), and there are less significant differences among the best beat trackers. Moreover, for DatasetSMC there is no set of best beat trackers, because all except four beat trackers are among the best performers for at least one measure. The performance of some beat trackers is close to the mean performance of an entirely deterministic (baseline) beat sequence, fixed at 120 bpm and generated as in Davies et al. (2009a). In general, this proves that the compiled dataset is more difficult for automatic beat tracking than Dataset1360, and again supports the validity of our proposed BT-MMA method.

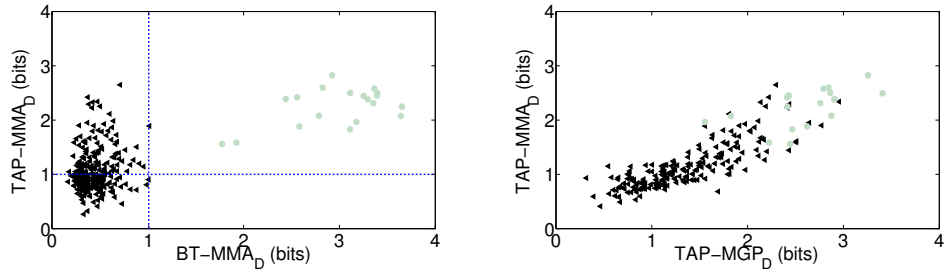
4.1.2. Perceptual vs. automatic beat tracking difficulty

Assessing perceptual difficulty

To better understand the difficulty of beat tracking, subjective listening aspects should be taken into account. In DatasetSMC, we can gain insight into these subjective aspects by using the spontaneous taps collected in the annotation process.

During the annotation of DatasetSMC, we were able to confirm that tapping spontaneously to an unknown piece without a clear and simple beat is a very

demanding process. Thus, we assume that perceptually easier files result in tap sequences that show a higher mutual agreement, analogous to the beat tracker outputs. In order to differentiate these agreements from the *MMA* obtained from beat trackers (i.e. BT-MMA) we will refer to them as **TAP-MMA**, and to the mean performance of the taps compared to ground truth as **TAP-MGP** (in contrast to BT-MGP). Figure 4.2a shows a scatter plot of these TAP-MMA$_D$ values against the *MMA* values between the chosen five beat tracking algorithms (BT-MMA$_D$). While the sparse cluster in the upper right corner indicates that high agreement of beat sequences implies high agreement of spontaneous taps, such a relation does not exist for low BT-MMA$_D$. In this case, we can observe the existence of a wide range of TAP-MMA$_D$ values. This implies that among files that are difficult for automatic beat tracking, there are difficult as well as easy files for the human tappers. In Figure 4.2b a high correlation between TAP-MMA$_D$ and the mean performance of the taps against the ground truth annotations (TAP-MGP$_D$) can be observed. This correlation supports the assumption that high agreement between subjects implies perceptually easier pieces. Comparing Figures 4.1b and 4.2b, we can see that in Figure 4.2b there are no separate clusters of data for very low *MMA* and MGP values. This indicates that, for the difficult samples, the human taps tended to be more accurate when compared to the ground truth, and that the spontaneous taps are characterized by a higher mutual agreement than the beat tracker outputs.



(a) (b)

Figure 4.2: TAP-MMA$_D$ and TAP-MGP$_D$ for annotated 217 files in DatasetSMC. Pieces which are supposed to be easy according to their BT-MMA$_D$ are depicted by gray circles. (a): Scatter plot of TAP-MMA$_D$ versus BT-MMA$_D$, blue lines indicate chosen border for difficult files for beat tracking (vertical line) and human tappers (horizontal line) (b): Scatter plot of TAP-MMA$_D$ versus TAP-MGP$_D$

In conclusion, we can state that, even without ground truth available, it is possible to reliably detect samples where automatic beat tracking will fail, but among these files there will be both files that are perceptually difficult and files that are easy. Because our aim is to facilitate improvement in beat tracking, we want to focus on those pieces that have a perceivable beat but that make

beat trackers fail. These pieces are located in the top-left rectangle of Figure 4.2a, and we will now focus on the signal properties that differentiate them from perceptually difficult pieces which are located in the lower-left rectangle of Figure 4.2a.

Signal properties

The general signal properties encountered in DatasetSMC are summarized in the tags that were assigned during the annotation process. Figure 4.3 shows the number of occurrences of all tags for the 217 annotated pieces. The most prominent tag is *expressive timing*, which was applied when a sample changes in tempo in correlation with its melodic phrase or segment boundaries (Todd, 1989) as often happens in romantic music. Other prominent tags related to tempo were *slow tempo*, *gradual tempo change* (*i.e.* one stable tempo changes gradually to a different stable tempo) and *tempo discontinuity* (*i.e.* a sudden tempo change). This confirms that any kind of tempo changes cause trouble for beat tracking approaches, and adds the characteristic of having a slow tempo to the list of problematic tempo-related features. Furthermore, ternary meter as a characteristic of the metrical structure of the composition also lead the beat trackers to fail, which suggests that many approaches may be tailored to track music mainly in a $\frac{4}{4}$ time signature. Characteristics related to the instrumental timbres, such as *lack of transient sounds* and *quiet accompaniment* complete the picture of the problematic signal properties that make beat trackers fail into three groups:

1. Timing/tempo related
2. Time signatures
3. Lack of clear rhythmic onsets

The tag *none* was applied when none of the other tags fit to the properties of the signal, and its appearance is always related to the files with high BT-MMA$_D$, *i.e.* the 19 easy files in DatasetSMC.
Having obtained an overview of the signal properties that make automatic beat tracking difficult, we would like to know which of these properties makes tapping for human listeners difficult. We want to address the question of whether the files in the upper and lower left rectangles of Figure 4.2a differ according to their signal properties. If we can identify some significant differences here this can give valuable insight into how to discriminate between perceptually difficult pieces and those that are difficult only for automatic beat tracking. To this end, features describing those discriminant signal properties might be used in a machine learning approach to automatically classify into one of the two classes. A threshold was set to a TAP-MMA$_D$ value of 1 bit (dotted horizontal line in Figure 4.2a), *i.e.* the same threshold that was applied to BT-MMA$_D$

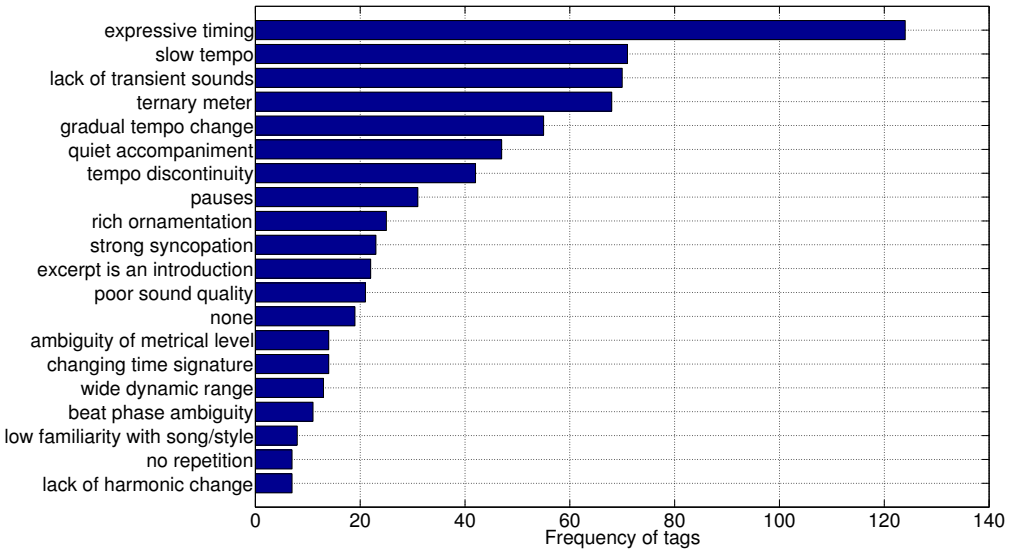


Figure 4.3: Frequency of tags for all annotated files in DatasetSMC. Tags indicate which signal properties made a sample appear difficult during the manual annotation.

when choosing difficult files for annotation. Then, a set of t-tests was applied in order to look into if the beat-annotated samples in the lower and upper left rectangles differed regarding their given tags. By performing this set of t-tests we can infer which properties lead to inaccurate tappings.

Table 4.2: Tags with different mean according to t-test, sorted by increasing p-value, from top to bottom. The presence of a tag implies that it appears significantly more frequently for low TAP-MMA$_D$

T-test: TAP-MMA$_D$	**p-value**
changing time signature	0.0010
expressive timing	0.0011
quiet accompaniment	0.0035
no repetition	0.0047
low familiarity with song/style	0.0110
beat phase ambiguity	0.0360

The results of the t-tests are listed in Table 4.2. For convenience, the appearance of a tag in the list means that it is significantly more present in files with low TAP-MMA$_D$. We can see that a change in time signature was the most important factor that leads to low tapping agreement. However, this tag is quite sparse among the dataset as can be seen in Figure 4.3. The most

prominent factors, taking into account their number of appearance, are expressive timing and quiet accompaniment. Hence, these factors apparently cause problems both for beat trackers and for human tappers. The list of properties given in Table 4.2 can serve as a guideline to which signal descriptors might be applied when trying to exclude signals from automatic beat tracking because of their high complexity even for human listeners. It is apparent that *e.g.* processing music with highly expressive timing should be postponed, as its beat is too complex to be spontaneously tracked even by human listeners. Demanding a proper beat tracking on music of this kind would resemble demanding high word recognition rates from an automatic speech recognizer in signals that cannot be perceived by a human listener. A first step may be to focus on music characterized by *e.g.* ternary meters, slow tempo or soft onsets, or other characteristics that do not impose increased difficulty to the human beat perception.

4.2. Voice suppression algorithms as a preprocessing step

As a result of the analysis of the general signal properties that makes beat tracking difficult, songs with strong and expressive vocals resulted in beat estimation errors even in the presence of a rhythmically stable accompaniment. This section focuses on beat estimation in this particular context and is motivated by previous research that showed the advantage of source separation techniques as a preprocessing step for automatic tempo estimation (Alonso et al., 2007; Chordia & Rae, 2009) and beat tracking (Gkiokas et al., 2012; Malcangi, 2005; Zapata & Gómez, 2012). We evaluate and discuss how voice suppression techniques improve rhythmic saliency in songs with highly predominant vocals and quiet accompaniment, and thus facilitate the automatic estimation of beat positions.

Source separation for improving tempo accuracy estimation has been proposed by Alonso et al. (2007), using harmonic + noise decomposition of the audio signal. To improve beat/tempo estimation, Gkiokas et al. (2012) use a percussive / harmonic blind source separation and Chordia & Rae (2009) use a blind source separation technique using Probabilistic Latent Component Analysis (PLCA). In this section we propose the use of source separation for voice suppression in excerpts with highly predominant vocals, in order to improve beat tracking performance. To the best of our knowledge, such an approach has not been previously considered.

In this study, we evaluate the performance of five state-of-the-art beat tracking algorithms in combination with seven different voice suppression approaches and a simple low pass filter. We consider an annotated dataset of difficult audio song excerpts with highly predominant vocals.

The Material of this section was previously published in Zapata & Gomez (2013)

4.2.1. Music material

Two datasets have been considered for this study, the first one is the *Dataset1360* (section 3.2) and the *DatasetSMC*[2] obtained in section 4.1. The difficulty of the song excerpts in *Dataset1360* and *DatasetSMC* was further assessed from the mean performance of the five considered beat trackers using the Mean Mutual Agreement proposed in Section 3.2.4. From the difficult excerpts, we finally selected 75 examples with highly predominant vocals (*DatasetVocal*).

4.2.2. Voice suppression methods

Voice suppression methods remove the singing voice from a polyphonic music signal by means of source separation techniques. According to Gómez et al. (2012), there are three main approaches for singing voice separation methods: spectrogram factorization, pitch-based inference and repeating-structure removal. In this study, we consider a set of state-of-the-art algorithms based on those different principles which are accessible for evaluation purposes. Three different spectrogram factorization approaches are evaluated. They are based on decomposing a magnitude spectrogram as a set of components that represent features such as the spectral patterns (basis) or the activations (gains) of the active components along time (Durrieu et al., 2011; Gómez et al., 2012; Marxer et al., 2012).
We also evaluate the use of four repeating-structure removal methods (Liutkus et al., 2012; Rafii & Pardo, 2012, 2013) which rely on pattern recognition to identify and extract accompaniment segments, without manual labeling, which can be classified as repeating musical structures. Finally, we evaluated the use of a low pass filter to remove higher spectral components in order to compare the results of voice suppression algorithms with a simple approach. We provide a brief description of the algorithms.

1. **Low Pass Filter (LPF):** Based on Masataka Goto et al. (1999), a simple Butterworth double-pole low-pass filter at 261.6 Hz (4800 cent) and Q = 0.707 was used as a baseline approach to remove high spectral components where the voice is assumed to be predominant [3].

2. **Instantaneous Mixture Model (IMM):** Durrieu et al. (2011) propose a source/filter signal model of a mixed power spectrum as a decomposition into a dictionary of pre-defined spectral shapes, which provide a mid-level representation of the signal content together with some timbre information. A non-negative matrix factorization (NMF) technique is used for source separation [4].

[2] http://smc.inescporto.pt/research/data/
[3] sox in.wav out.wav lowpass 261.6
[4] www.durrieu.ch/research/jstsp2010.html VU output

3. **Low Latency Instrument Separation (LLIS):** This method allows voice suppression under real-time constraints, and it is based on time-frequency binary masks resulting from the combination of azimuth, phase difference and absolute frequency spectral bin classification and harmonic-derived masks. A support vector machine (SVM) is used for timbre classification, and for the harmonic-derived masks, a pitch likelihood estimation technique based on Tikhonov regularization is used. We refer to (Marxer et al., 2012) for a detailed description of the algorithm.

4. **Repeating Pattern Extraction Technique (REPET):** REPET[5] is a method for separating the repeating background from the non-repeating foreground in an excerpt audio mixture. The approach assumes that musical pieces are often characterized by an underlying repeating structure over which varying elements are superimposed. The system identifies the repeating elements in the audio, compares them to repeating models derived from them, and extracts the repeating patterns via time-frequency masking. REPET with sliding window (REPET win) is an extension of the algorithm to full-track songs that applies the algorithm to local sections over time by using a fixed sliding window. We refer to (Rafii & Pardo, 2013) for a detailed description of the algorithm.

5. **Adaptive REPET (REPET ada):** The REPET method is originally intended for excerpts with a relatively stable repeating background. For full-track songs, the repeating background is likely to vary over time, so the adaptive REPET can be directly adapted along time by locally modeling the repeating background to handle varying repeating structures. This method is detailed in (Liutkus et al., 2012).

6. **REPET with Similarity Matrix (REPET sim):** This method by Rafii & Pardo (2012), generalizes the REPET approach to handle cases where repetitions also happen intermittently or without a fixed period, thus allowing the processing of music pieces with fast-varying repeating structures and isolated repeating elements. Instead of looking for periodicity, this method uses a similarity matrix to identify repeating elements. It then calculates a repeating spectrogram model by using the median and extracts repeating patterns using a time-frequency masking.

7. **Singing Voice Separation (UJaen):** The last approach considered, described by Gómez et al. (2012), factorizes a mixture spectrogram into three separated spectrograms (Percussive, Harmonic and Vocal). Harmonic sounds are modeled by sparseness in frequency and smoothness in time, percussive sounds by smoothness in frequency and sparseness in time and vocal sound are modeled by sparseness in frequency and sparseness in time. A predominant f_0 estimation method is used for the vocal

[5] music.cs.northwestern.edu/

separation, for which the vocal parts were previously labeled by hand. The implementation used in this study had the same source separation method, but was completely unsupervised.

4.2.3. Beat trackers

To analyzes the effect of the voice suppression in the beat tracking of audio signal with high predominant vocals we consider five beat trackers, four state-of-the-art beat tracking approaches: Beatroot (Dixon, 2007), Degara (Degara et al., 2012), IBT (Oliveira et al., 2012), Klapuri (Klapuri et al., 2006). and the multi feature beat tracker (Zapata et al., 2012a).

4.2.4. Evaluation measures

Among all of the proposed evaluation metrics, we consider the most permissive continuity measures that Allowed Metrical Level errors, because it considers that beat estimations at double or half of the correct metrical level are valid, and it also accepts off-beat estimations. We compute AMLc (Allowed Metrical Level with continuity required) and AMLt (Allowed Metrical Level with no continuity required) as defined in (Hainsworth & Macleod, 2004; Klapuri et al., 2006). Output range between [0 - 100]%.

4.2.5. Results

Table 3.6 shows the average evaluation results of the considered beat tracking systems on Dataset1360 and Table 4.3 shows their performance on the DatasetVocal. Beat estimations and evaluation data are publicly available [6].
We observed that the beat tracking performance drastically decreases for songs with highly predominant vocals for all the considered methods. This confirms our hypothesis and the observations in Section 4.1.2, which identified the difficulty of these examples. To get an idea of the best algorithmic performance currently achievable, we define an “Oracle" beat tracker whose performance is equal to the best performance obtained for each excerpt by any of the considered algorithms. For the *DatasetVocal*, the Oracle tracker would yield 33.95% and 52.65% accuracy for the AMLc and AMLt measures respectively. Evidently, there is still much room for improvement for this type of music.
Regarding the advantage of using voice suppression techniques, we observe that all beat trackers increases their mean performance (AMLc and AMLt measures) over *DatasetVocal* when using *UJaen* and *IMM* as a preprocessing step, although the accuracy increase is small. In addition, Degara's beat tracking approach (with one of the highest performance in *Dataset1360*) statistically improves its performance for all the evaluated voice suppression algorithms ($p<0.05$). Moreover, all beat trackers improve their accuracy (AMLt

[6] sites.google.com/site/tempoandbeattracking/

Table 4.3: Mean AMLc and AMLt performance results in the original and the processed audio files from *DatasetVocal* per beat tracking system (* indicates statistically significant improvements with $p<0.05$)

Measure	BT name	Original Audio	LPF	Repet	Repet ada	Repet sim	Repet win	LLIS	UJaen	IMM
AMLc (%)	**Beatroot**	10.54	10.97	9.14	7.59	10.12	9.37	10.49	**14.68**	11.97
	Degara	16.88	24.17*	24.94*	25.38*	24.13*	24.86*	23.91*	26.23*	**26.83***
	IBT	24.70	17.07	16.51	20.00	18.39	22.02	19.75	24.79*	**26.46***
	Klapuri	22.61	24.52	25.61	22.53	24.53	22.74	23.37	**29.44**	26.31
	MultiF_inf	21.32	24.64	28.11	27.26	23.87	22.92	27.38	**29.47**	28.75
AMLt (%)	**Beatroot**	25.39	25.19	24.84	19.72	26.17	23.38	27.23	**31.89**	27.36
	Degara	28.70	37.64*	38.45*	37.67*	37.99*	38.28*	37.86*	40.13*	**42.24***
	IBT	27.55	37.45	27.35	32.38	29.55	33.17	32.78	39.40*	**40.49***
	Klapuri	36.60	39.12	38.43	36.41	38.70	34.49	39.43	**43.96**	43.02
	MultiF_inf	34.88	37.34	40.99	39.59	38.45	35.85	41.51	41.75	**42.81**
Process Time [=] Min			0.37	3.42	15.54	14.30	6.74	221.54	293.51	14723.12

Table 4.4: Percentage of songs that improves and degrades in each voice suppression system

Audio Files	Measure	LPF	Repet	Repet ada	Repet sim	Repet win	LLIS	UJaen	IMM
Improved (%)	AMLc	8	1.33	4	4	4	9.33	10.66	6.66
	AMLt	12	4	14.66	8	5.33	12	13.33	13.33
Degraded(%)	AMLc	1.33	4	2.66	2.66	1.33	8	4	1.33
	AMLt	5.33	8	2.66	1.33	4	2.66	5.33	2.66

measure) by using *LLIS* as a preprocessing step. Finally, the three best performing methods on *Dataset1360* experience an increase of the performance on *DatasetVocal* using very simple (*LPF*) and fast (*REPET*) approaches.
One of the most critical aspects of using voice suppression over large collection is the computational cost (The runtime is provided in Table 4.3). Although these approaches vary in terms of optimization level, we observe large differences in runtime (e.g.*IMM* is almost 50 times slower than *UJaen* algorithm).
In Table 4.4 we present the total number of songs for which all beat trackers obtained improved performance when using voice suppression algorithms. We observe that the performance is improved for the majority of songs (with the exception of the *REPET* method). We also observe that the better the performance of the voice suppression algorithm, the greater the increase in beat tracking performance.
If we apply voice suppression methods not only to music with highly predominant vocal but to *Dataset1360*, we only get small improvements in accuracy for the combination of all *REPET+Degara*, *LLIS+Klapuri* and *REPET sim+ IBT*. None of these improvements are statistically significant, though. We then conclude that while voice suppression might be beneficial for excerpts with highly predominant vocals, these algorithms do not provide enhancements for varied datasets.
Voice suppression allows beat trackers to achieve higher estimation accuracy than the Oracle in some song excerpts with highly predominant vocals, because they enhance the signal and allow a better mid-level representation for beat tracking. Although the highest increase is yielded by the *IMM* voice suppression method, this approach needs a very high computation time (around 196 min per song) to process the audio. Other methods such as *LLIS* and *UJaen* yield similar results in less time (around 3.9 min per song). This fact makes them more suitable to process large music collections.

4.3. Automatic beat annotation in large datasets

In Section 4.1 we present a technique to automatically identify challenging examples for beat tracking without the need for ground truth annotations. The technique is based on measuring the mean mutual agreement (*MMA*) between a committee of state of the art beat tracking algorithms, where low mutual agreement (or put another way, high disagreement) between beat outputs was shown to be a good indicator of low performance against the ground truth. We empirically determined an *MMA* "failure" threshold below which beat tracking performance was shown to be very poor, and created a new database comprised of challenging songs with *MMA* below this threshold.

This section is based upon work in collaboration with Andre Holzapfel, Matthew E. P. Davies, João Lobato Oliveira and Fabien Gouyon. This work was published in the ISMIR 2012 peer reviewed conference, Zapata et al. (2012b)

In this section we address the opposite issue, where, instead of trying to find where beat tracking algorithms fail, we wish to identify where beat tracking has been successful. When ground truth annotations are available this question can be easily answered, however the problem is when no ground truth exists, *i.e.*, in the vast majority of music. The current means for doing so is simply to extrapolate the performance on the limited dataset, for which a precise evaluation can be conducted, and assume that this is representative.
In light of our previous concerns about the make-up of these annotated databases, we believe that extrapolating performance in this way and expecting reliable results will be overly optimistic. Therefore when seeking to determine an unbiased measure of performance we can either manually annotate more and more music examples for evaluation, or attempt to estimate beat tracking performance without ground truth. Due to the impractical nature of the first option, we pursue the second. Furthermore, if no ground truth is required, then performance can be estimated on very large (effectively unlimited) collections of music.
We attempt to determine an *MMA* "success" threshold above which we can have high confidence in the beat tracking output of a committee of state of the art algorithms. We determine the success threshold by means of a subjective listening test, where listeners are asked to rate the quality of the beat output given by the committee across a range of songs for which the *MMA* has been calculated. In each case the beat tracker output chosen to represent the committee is selected automatically as the one which most agrees with the remainder of the committee, *i.e.*, the beat tracker output with the maximal mutual agreement (***MaxMA***). We demonstrate (Section 3.2.5) that selecting between beat tracker outputs using *MaxMA* leads to improved performance over consistently picking any individual algorithm from the committee.

4.3.1. Beat tracking annotation

Having illustrated the validity of using the *MaxMA* method to select a beat tracker output among a committee of algorithms on a manually annotated dataset, we now turn our attention to applying it to a large collection of non-annotated data. For very large collections it is impractical to expect there to be ground truth annotations on which to base the performance evaluation. To understand how well the state of the art in beat tracking can automatically annotate beats in large collections we employ our *MMA* and *MaxMA* methods and attempt to determine the proportion of songs for which the beat estimates are acceptable via a subjective listening test. We want to establish a threshold on *MMA* above which the beat tracker outputs are perceptually acceptable. For each file, the beat tracker output will be chosen using the *MaxMA* method. In order to correlate the *MaxMA* technique to automatic annotated a large dataset and determine the perceptual success of the beat estimation, we used the least stringent continuity-based measure, AMLt (Allowed Metrical Level

with no continuity required), where beats are accurate when consecutive falling within tempo-dependent tolerance windows around successive annotations. Beat tracker outputs are also considered accurate if beats occur on the off-beat, or are estimated at double or half the annotated tempo. This performance measure provides a more intuitive scale of 0 to 100% than Information Gain and allows some ambiguity in the choice of metrical level at which the beats are estimated.

Million song subset

The large collection we aim to automatically annotate is the **MillionSongSubset** from the Million Song Dataset (Bertin-Mahieux et al., 2011). The subset is comprised of 10,000 songs without ground truth for which audio previews were obtained. The majority of audio previews were either 30 sec or 60 sec in duration, however to provide sufficiently long excerpts for beat tracking we discarded those shorter than 20 sec. This left a set of 9940 songs on which to automatically estimate beats. To complement the audio data, we obtained 31696 Last.fm[7] tags which covered a subset of 4638 songs.
Once all of the audio and meta data was collected we ran the committee of beat tracking algorithms recording the *MMA* value per excerpt and saving the *MaxMA* beat tracker output on the 9940 songs.

Subjective listening test

The aim of our listening test was to determine an *MMA* threshold above which the beat tracker output given by the *MaxMA* method was deemed acceptable to human listeners. By subsequent inspection of the number of songs in the dataset above this *MMA* threshold we could then estimate the proportion for which beat tracking can be considered successful.
Just as it is not possible to hand annotate beats in nearly 10,000 songs, it is equally impractical to ask participants to listen and rate this large number. As alternative to the exhaustive rating of all audio songs, we selected 8 levels of $MMA = [0.5, 1.0, 1.5, \ldots, 4.0]$ bits and chose the 6 closest songs from the MillionSongSubset to each *MMA* level, giving a total of 48 songs to summarize the dataset. To create the musical stimuli for the listening test we constructed stereo audio files containing a mixture of source audio and the *MaxMA* beat output synthesized as short click sounds. To mitigate the effect of errors in beat tracking at the start of songs, which might bias the listener ratings, each musical stimulus was formed out of the middle 15 s of each song. To allow listeners to hear the audio with and without click sounds, we panned the source audio on its own on the left channel, and on the right channel we mixed the click sounds conveying the beats with a quiet version of the source audio.

[7] http://labrosa.ee.columbia.edu/millionsong/lastfm

Through informal listening tests prior to the main experiment, this was deemed an acceptable method for creating the stimuli.
We recruited 25 participants to take the listening test (21 male, 4 female) with an age range of 23 to 41 (mean = 31 years, std = 4.7 years). The participants' level of music training ranged from 0 to 20 years (mean = 8.7 years, std = 7.7 years). Each participant was instructed to perform the test in a quiet environment with good quality headphones. Prior to starting the main test, the participants were given three training examples (not in the main set of 48). The training phase was used for three reasons: *i)* to familiarize participants with the type of musical stimuli in the test, *ii)* for the participants to understand the panning of the beats in the stimuli and *iii)* so the participants could set the playback volume to a comfortable level. To prevent order effects in the stimuli, each participant was given an individual playlist of songs in a different random order.
In taking the test, the participants were asked to answer the following question: "*How do you rate the overall quality of the given click as a beat annotation of the piece?*" The options for rating were: 1 - Bad, 2 - Poor, 3 - Fair, 4 - Good, 5 - Excellent.

4.3.2. Results

Listening test

Figure 4.4 presents a comparison between the human ratings and the *MMA* of our committee of beat trackers for the selected 48 pieces of the MillionSong-Subset. The plot shows that for an *MMA* equal to 1.5 bits the mean rating was 3.7 (Good) with a standard deviation of 0.93. However, for *MMA* equal to 1 bit, the mean rating was much lower, at around 2.4 (Poor). Performing a t-test, we found the difference between the mean ratings at these *MMA* values to be highly significant ($p < 0.0001$). On this basis we can easily identify an *MMA* threshold of 1.5 bits which separates perceptually acceptable beat tracking from inaccurate beat tracking.

MMA Threshold

By selecting an *MMA* of 1.5 bits as a threshold of perceptual confidence for beat tracking we found 996 songs (73%) in Dataset1360 and 7252 songs (coincidentally also 73%), in the MillionSongSubset above this limit (see Figure 4.5). Table 4.5 shows the AMLt scores for the Oracle, *MaxMA*, Best Mean, and MinMA for the two subsets of Dataset1360 separated by *MMA* = 1.5 bits, evaluated against the ground truth. The beat tracking performance is consistently high for songs with *MMA* >1.5 bits, with a mean *MaxMA* performance of ≈90%, which must be considered very accurate, and hence hints at a meaningful relationship between subjective judgment of beat tracking and the AMLt scores obtained from the objective evaluation. While beat tracking performance is

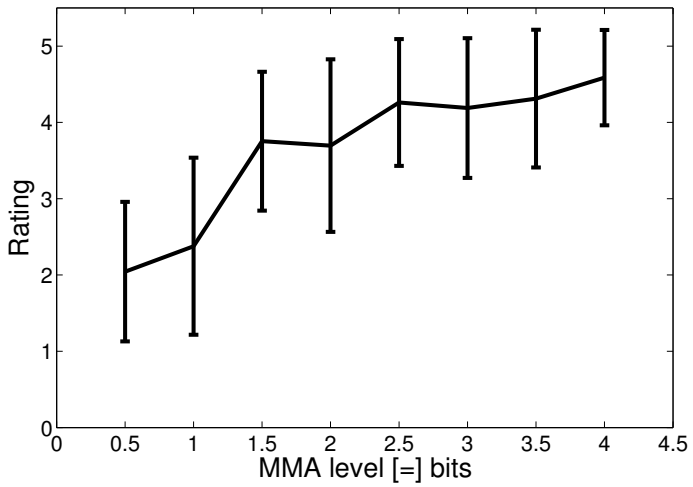


Figure 4.4: Listening test ratings *vs MMA* for the selected 48 music excerpts, from the MillionSongSubset.

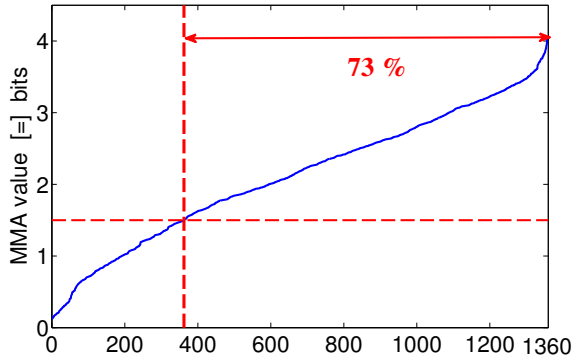
Name	AMLt (%)	*MMA*
Oracle	95.4	*MMA*>1.5
MaxMA	89.9	
Best Mean	86.3	
MinMA	63.9	
Oracle	70.9	*MMA*<1.5
MaxMA	58.8	
Best Mean	54	
MinMA	50.1	

Table 4.5: Mean AMLt score of Oracle, *MaxMA*, Best_Mean, and MinMA for the two subsets of Dataset1360 divided by an *MMA* threshold of 1.5 bits.

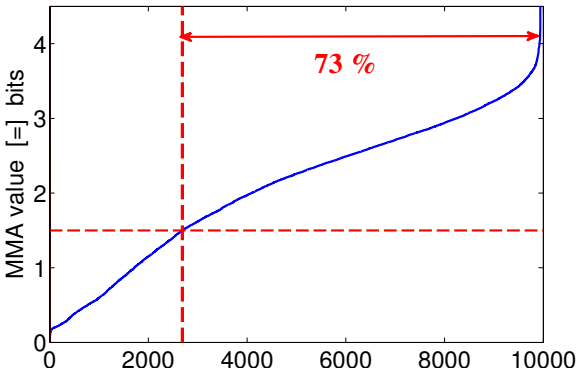
lower for $MMA < 1.5$ bits this does not mean the *MaxMA* beat estimations cannot be perceptually accurate, merely that we do not have high confidence in them.

Last.fm Tag analysis

Given the *MMA* threshold and collected Last.fm meta-data, we now look at the genre-related tags of the songs that appear significantly more often (with $p < 0.0001$) in the MillionSongSubset with *MMA* above and below 1.5 bits. These are shown in Table 4.6. From inspection of the table we can see that the genres above the *MMA* threshold are those which we would typically associate with being "easier" for beat tracking where as those below the threshold appear more challenging. Seeing all genre labels related to metal music below



(a) Dataset1360



(b) MillionSongSubset

Figure 4.5: Datasets sorted by *MMA* and the perceptual threshold = 1.5 bits.

the threshold was a surprising result since this music is strongly percussive and is not characterized by wide tempo changes. The fact that metal music consistently falls below the threshold indicates it might be the "noisy" element of the music which causes it to be difficult. To the best of our knowledge we are unaware of many metal examples in existing beat tracking databases. This suggests it is something of a forgotten genre for beat tracking.

Another important observation is related to the tag frequency for genre labels above and below the threshold. There is a far higher proportion of songs tagged "Rock" and "Pop", and in general the tags used above the threshold appear much more frequently than those below it. From this we can infer that, just as Dataset1360 is biased towards easier cases for beat tracking, the same could be said of the MillionSongSubset. Evidence for this conclusion can be found in the description of the MillionSongDataset itself (Bertin-Mahieux et al., 2011) where the lack of diversity is mentioned; in particular the small amount of classical and world music.

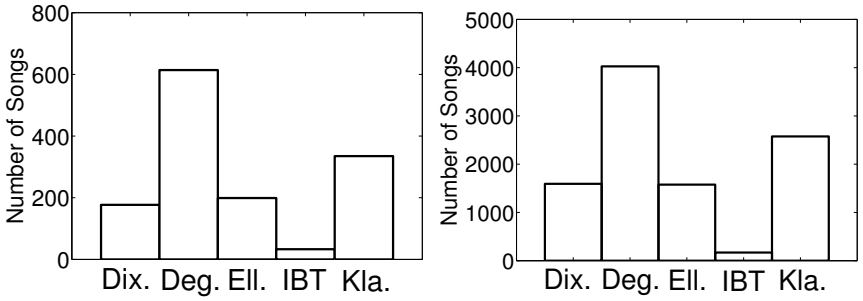
Given the disproportionate number of easier songs for beat tracking in this dataset, our estimate of 73% of songs for which beat tracking is acceptable may still be an optimistic estimate of the true level of beat tracking performance across all music.

Table 4.6: Frequency of the genre-based occurrence of tags for the two subsets of MillionSongSubset divided by an *MMA* threshold of 1.5 bits.

Tag	**Frequency**	***MMA***
Rock	1080	
Pop	680	
Dance	320	
Hip-hop	271	*MMA*>1.5
Rap	193	
Pop rock	154	
Reggae	149	
Jazz	227	
Instrumental	199	
Death metal	80	
Black metal	74	*MMA*<1.5
Progressive metal	59	
Classical	36	
Grindcore	28	

MaxMA choice of beat tracker

Having explored the main results of applying *MaxMA* to automatically annotate beat locations, we now address the properties of the committee. Figure 4.6 presents histograms for both evaluated datasets depicting the proportion of songs where each beat tracking algorithm is selected as the *MaxMA* beat output. Both histograms show similar shapes, indicating that there may be similar properties between the musical content of both datasets. The two most chosen algorithms are those of Degara and Klapuri; both of which perform most accurately against the ground truth, and can be considered the best among the state of the art methods. As to why the Degara algorithm is chosen more frequently than that of Klapuri, results in Degara et al. (2012) indicate that the inter-quartile range of the Degara algorithm is smaller than that of Klapuri (for a similar median), implying it is "wrong" in a lower proportion of songs.



(a) Dataset1360 (b) MillionSongSubset

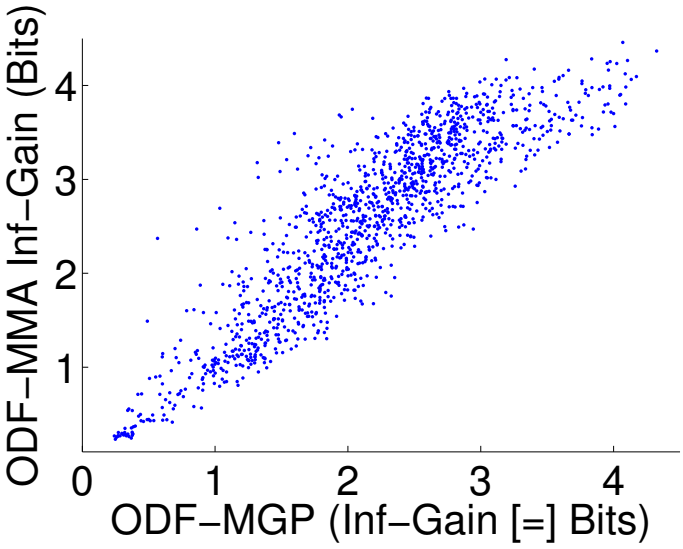
Figure 4.6: Histograms with the number of times each algorithm is chosen with the *MaxMA* approach.

4.4. Multi Feature Mean Mutual Agreement and confidence threshold

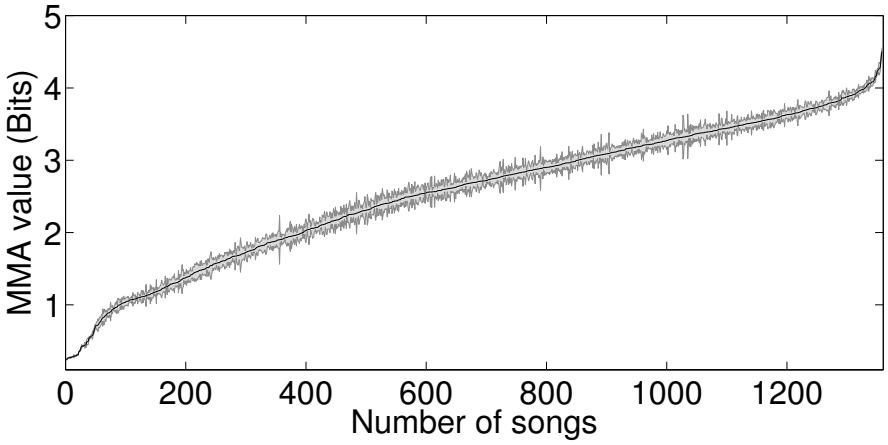
To further explore the properties of the multi-feature beat tracker, we undertake an analysis of the *MMA* values. First, we seek to recreate the primary result from section 3.2.5 which showed a high correlation between the *MMA* of the beat tracking committee and the mean performance of the committee against ground truth, the MGP. As shown in Figure 4.7a we can see that the *MMA* (using Information Gain) is strongly correlated with the MGP of the committee using the set of ODFs. Thus we can confirm that disagreement between the beats of the committee is indicative of overall poor beat tracking accuracy and vice-versa.

While we could not find a statistically significant difference in performance between the six member and nine member committees, we would like to explore the extent to which the mutual agreement changes based on the number of committee members. To this end we show the range of observed *MMA* values obtained with committees of six, seven, eight and then nine members, in Figure 4.7b. As expected, we find very low variance in the *MMA* values obtained with committees of different sizes both when the mutual agreement is very low and likewise when it is very high. In this sense the variation in the size committee becomes apparent in the middle *MMA* range.

To complete our analysis of *Dataset1360*, we investigate whether we can use the *MMA* vs MGP correlation to automatically assign either high or low confidence to the estimated beats. Following the section 4.3.2, where a threshold of *MMA*>1.5 bits was found to be indicative of acceptable beat tracking, we re-examine the beat tracking performance on songs with *MMA* above and below 1.5 bits. As shown in Table 4.7, we see that performance is far higher for excerpts where the *MMA* is above the threshold compared to below it. Of the 1360 excerpts in the dataset, we found 1126 (82.9%) were above it, for which



(a)



(b)

Figure 4.7: (a) ODF mean mutual agreement (MMA) vs ODF Mean ground truth performance(MGP). (b) Error-bar of *MMA* calculated with 6 and more committee members vs songs, sorted by *MMA*(9 committee members)

the **AMLt** value >86% for all configurations of the multi-feature beat tracker. While beat tracking performance is lower for $MMA < 1.5$ bits this does not mean the multi feature beat estimations cannot be accurate, merely that we do not have high confidence in the result, likewise there will be cases with *MMA* above the threshold which are not accurate. These can arise when the beats are tapped at a meaningful metrical level, but one which is not included with in the set of allowed levels specified for **AMLt** (Davies et al., 2009a).

Table 4.7: Mean scores (%) of Oracle, committee of 5 beat trackers, multi-feature beat tracker and best mean performance beat tracker (BestBt) for the two subsets of *Dataset1360* divided by an *MMA* threshold of 1.5 bits.

Name	CMLc	CMLt	AMLc	AMLt	MMA
Oracle	70.4	72.8	91.8	94.4	>1.5
5BT	52.6	56.1	80.4	87.5	
Multi-Feature Regularity	53.3	56.1	81.3	86.7	
Multi-Feature InfGain	51.4	54.8	80.4	86.6	
MultiFt Essentia	51.7	55.0	80.0	86.3	
BestBt	53.5	57.4	77.6	84.0	
Oracle	38.5	50.8	54.8	71.8	<1.5
Multi-Feature Regularity	21.7	32.8	35.6	53.4	
Multi-Feature InfGain	24.7	35.5	36.3	52.8	
5BT	19.1	29.5	32.7	51.8	
MultiFt Essentia	19.2	29.8	32.7	51.5	
BestBt	19.9	29.8	31.7	47.0	

4.5. Conclusion and future work

In this chapter, we presented a method based on mutual agreement of beat sequences to detect informative samples in non-annotated data collections. We compiled and annotated a new dataset that consists mainly of pieces with low mutual agreements, and showed that this dataset is significantly more difficult for state of the art beat tracking algorithms than the largest existing collection. Using the new difficult dataset, we analyzed the signal characteristics that make beat trackers fail, and research the extent to which these characteristics coincide with the properties that make tapping difficult for humans.

The proposed method of measuring mutual agreement represents an efficient approach to improve diversity in existing datasets, as well as a simple technique for improvement of beat tracking in large non-annotated datasets. While all the directions for future work have so far been related to beat tracking, we strongly believe that, given suitable evaluation metrics, Our method based on *MMA* and *MaxMA* could be readily applied in other contexts, *e.g.* to detect problematic files in chord recognition or onset detection where it may be valuable to reject the use of beat tracking as a temporal analysis component. We therefore encourage MIR researchers to explore its usage in problems such as onset detection, chord detection,structural segmentation, and music transcription.

Based on our informal analysis of human tapping it appears that expressive timing contributes strongly to making music difficult to tap to. Furthermore it may not be musically appropriate to attempt to follow large expressive changes precisely. The musical experts who assisted in the annotation process demonstrated that more musically meaningful annotations could be obtained by tapping a stable pulse around which the timing changes deviate. However this level of tapping required extensive musical training (beyond the level of the authors) and provides strong evidence towards rejecting beat tracking for musical pieces of this nature. For more realistic advances in beat tracking, we propose investigating techniques for music with properties that do not pose considerable difficulties for humans, such as pieces characterized by ternary meter, slow tempo, or soft instrument onsets. In order to reliably detect difficult samples using mutual (dis-)agreement, we demonstrated that the choice of the evaluation measure is crucial, and that the Information Gain was better suited to this task than both the F-measure and AMLt evaluation methods. However, the Information Gain method appears less effective in highlighting where beat tracking algorithms strongly agree with each other. Hence in future work we will explore methods to combine the information from different evaluation methods.

We have demonstrated that voice suppression techniques push up the glass ceiling of state-of-the-art beat tracking algorithms in music with highly predominant vocals. Beat trackers seem to benefit more from voice suppression in difficult songs with highly predominant vocals and voice suppression can be used as a pre-processing stage without having to modify the beat tracking algorithm. Nevertheless, this approach would decrease beat tracking performance in the other situation, i.e. *a capella*, choral or music where the voice carries relevant rhythmic information. Future work has to be devoted to automatically selecting the candidate material where voice suppression would have a positive effect on beat tracking, additionally use full length stereo songs in order to evaluate voice suppression methods in more realistic setting, because most of the voice suppression algorithms use spatial information.

Through a subjective listening test we determined an *MMA* threshold between this committee of beat trackers of 1.5 bits above which we believe automatic beat tracking can be applied with high degree of confidence. Based on this perceptual confidence, we demonstrate that around 73% of the MillionSongSubset could be automatically annotated using our committee of beat trackers. This proportion of songs for which we can be confident in an automatic beat annotation was also verified in a second dataset with manually annotated ground truth. Given the apparent bias in these datasets towards easier genres for beat tracking, we consider this value of 73% to be somewhat optimistic. As future work this hypothesis can be to verify by measuring *MMA* in more diverse datasets.

Regarding the types of music which formed the remaining 27% of the MillionSongSubset (*i.e.*, those below the threshold) we found a high proportion of tags related to metal and similar "noisy" styles of music. Beyond classical music and jazz, which are known to be challenging for beat tracking systems, we consider the difficulty of beat tracking in Metal genre to be a new and unexpected result, and furthermore an interesting area for the future development of beat tracking algorithms.

As well as the five beat tracker committee, the Multi feature beat tracker estimates the confidence of beat tracking and select the best beat tracking output without ground truth annotations, using a the committee of onset detection functions and measuring the Mean Mutual Agreement and the Maximum Mutual Agreement respectively. Additionally there is no statistical difference between the mean performance and the beat tracking confidence when compared to the five beat tracker committee.

One limitation of our approach in the *MMA* threshold test, may be the use of short song excerpts for the listening test. This was done to make the listening test as manageable as possible for a wide range of participants. Nevertheless, to obtain a greater understanding of subjective ratings for longer musical excerpts and a better understanding of perceptual difficulty in beat perception is better to conduct more sophisticated subjective listening experiments.

4.6. Summary

In this chapter, we present a method that can identify challenging music samples for beat tracking without ground truth. Our method, motivated by selective sampling, is based on the measurement of mutual agreement between beat sequences, and we show the influence of choosing different evaluation measures to compute this agreement. Using this approach, we demonstrated how to compile a new dataset, whose signal properties make beat tracking difficult, and examine this difficulty in the context of perceptual and musical properties. Based on tag analysis, we indicated musical properties which advances in beat tracking research would be most profitable. Additionally, we proposed the use of voice suppression systems to enhance the signal for a better mid-level representation for beat tracking in difficult songs with highly predominant vocals.

To automatically beat annotate and estimate the confidence of beat tracking in a dataset without ground truth annotations we have proposed the use of two methods based on the mutual agreement between a committee of beat tracking algorithms. The first, the Mean Mutual Agreement, was used to estimate the level of consensus between the beat outputs of the committee. The second, the Maximum Mutual Agreement, was used to select the best beat tracking output from the committee of beat trackers. Furthermore, we established a threshold for perceptually acceptable beat tracking based on the mutual agreement of a committee of beat trackers. In the first step we use an existing annotated dataset to show that mutual agreement can be used to select one committee member as the most reliable beat tracker for a song. Then we conduct a listening test using a subset of the Million Song Dataset to establish a threshold which results in acceptable quality of the chosen beat output. For both datasets, we obtain a 73% of trackable music, and we look into which data tags are related to acceptable and problematic beat tracking. The results indicate that current datasets are biased towards genres which tend to be easy for beat tracking.

The multi feature beat tracker and the committee of beat trackers, can automatically beat annotate, estimate the confidence of beat tracking and identify challenging music samples for beat tracking in a dataset without ground truth annotations using the measurement of mutual agreement between the committee members.

CHAPTER 5

Conclusions

In this thesis, we carry out an extensive comparative evaluation and combination of automatic rhythm description systems for tempo estimation and beat tracking from audio signals. We evaluated 32 tempo estimation and 16 beat tracking state of the art systems in order to identify their characteristics, and how they can be combined to improve the actual performance on these tasks. Moreover, we described a new method for automatic beat annotation with a confidence degree value of its beat tracking estimation. This confidence degree can identify challenging music samples for beat tracking in a dataset without ground truth annotations based on the measurement of mutual agreement between a committee of beat tracker systems. Based on this method, we compiled and annotated a new dataset that consists mainly of challenging pieces for state of the art beat tracking algorithms. Finally, we present a method for the extraction of beat times, based on a committee of multiple onset detection functions and one beat tracker model.

5.1. Thesis contributions

In order to guarantee the reproducibility of the results of this research,

- scientific papers
- data results
- built datasets

are publicly available at: `http://mtg.upf.edu/people/jzapata`

We now summarize the main outcomes of the research within this thesis.

Tempo estimation

We evaluated and compared 28 academic algorithms and 4 commercial approaches for tempo estimation, that we consider representative on current approaches. The best result, obtained by a beat tracker, can be enhanced by a heuristic decision tree combination with the other methods and their results are higher than each approach by itself. We found that the best performing algorithms share the following characteristics: frequency decomposition, periodicity detection prior to the multi-band integration, *tatum* detection and a post-processing block which reduces the number of double and half error tempo estimations. Furthermore, algorithms involving band decomposition and computes the periodicity detection before the multi-band integration achieve better results.

Improving beat tracking research

We evaluated 16 state of the art beat tracking systems and presented a method that can automatically identify challenging music samples for beat tracking without ground truth, based on the mutual agreement of a committee of beat trackers, selected from the evaluated systems, and computed using the Information Gain measure. Using this approach, we compile a new dataset, whose signal properties make beat tracking difficult, and examine this difficulty in the context of perceptual and musical properties. Based on tag analysis, we indicate that changing time signature, expressive timing, quiet accompaniment, no repetition and beat phase ambiguity are musical properties where advances in beat tracking research would be most profitable. The Dataset is available at `http://smc.inescporto.pt/research/data/`
Moreover, based on the mutual agreement between a committee of beat tracking algorithms we have proposed the use of two methods to automatically beat annotate and estimate the confidence of beat tracking in a dataset without ground truth. The first, the Mean Mutual Agreement (MMA), was used to estimate the level of consensus between the beat outputs of the committee, challenging songs have $MMA < 1$ bit. The second, the Maximum Mutual Agreement, was used to selecting the best beat tracking output from the committee of beat trackers and its mean AMLt performance is around 85% for songs with MMA > 1.5 bits.
Finally, to improve automatic rhythm description, our experiments suggest that voice suppression systems enhance the audio signal for a better mid-level representation for beat tracking in difficult songs with highly predominant vocals.

Beat tracker

We present an stand-alone beat tracker (the Multi feature beat tracker) that extends the work of the beat trackers committee, using a committee composed by multiple onset detection functions as inputs to one beat tracker model, and the final output is selected by *MaxMa*. Results of our experiments demonstrated how our approach outperforms (in the evaluated measures) the current state of the art beat trackers. From a statistically point of view, it is significant in most of the evaluated measures and it is comparable with the evaluation results of the five beat trackers committee. As well as the committee of beat trackers, the Multi-Feature beat tracker not only can automatically annotate beats but also it estimates the confidence of beat tracking and it identifies challenging music samples for beat tracking in a dataset without ground truth annotations using the measurement of *MMA*.
The Multi-Feature beat tracker is released under the GNU Affero general public license and is publicly available at:
`http://essentia.upf.edu/` (Bogdanov et al., 2013)
Algorithm: *BeatTrackerMultiFeature()*.

5.2. Future work and perspectives

The proposed methods for automatic rhythm description within this thesis can not be considered as a final solutions to the problems addressed. All the levels in the algorithms are potential areas for rhythm description improvement. We now outline some areas of potential future research extensions to our work.

Tempo estimation

Future work in tempo estimation could be devoted to build new public datasets for evaluation, compiling music pieces mainly by audio songs with signal characteristics whose make tempo estimators fail. The meter data, *tactus* and *tatum* of each song-excerpt could improve the analysis of the evaluation results, as well as it limits the errors inherent in the metrical level errors, because is not clear if the relations 2, $\frac{1}{2}$, 3 and $\frac{1}{3}$ are a metrical level error or of another kind of estimation error.
Most of the tested tempo estimators use ACF to find periodicities in the signal but the performance of these approaches are statistically different. On the other hand other systems used different pulse induction systems (e.g. ACF, AlonsoSP = spectral product, Scheirer = comb filter bank) and they had similar statistical performance. As a result, it is not clear which systems (e.g. ACF, comb filter bank, DFT, spectral product) or combination of these are appropriate for pulse induction. Moreover, the relation between audio features and pulse induction systems could be studied.

In addition, different methods are used to select the best tempo output on the periodicity signal (e.g. histogram, peak selection, tempo hypothesis with agents setup, Viterbi decoding and clustering) however, looking at the results of the evaluation, it is not noticeable which of these methods yields better results. For future work, a modular analysis of different audio features combination, periodicity functions and selection methods, such as the modular beat tracking evaluation by (Stark, 2011, chap. 4), would shed light on this relation for tempo estimation improvement.

Finally, the implementation of a single system that unifies the combination of different tempo estimation approaches could be done instead of using a heuristic combination of different tempo estimation systems. The performance of the heuristic tempo combination falls some way below, compared to the Oracle results. This suggests that there is still room to explore new combination methods and develop more accurate selections methods with the existing algorithms.

Beat tracking challenges

In the process of building the SMC dataset, the musical experts, who assisted in the annotation process, demonstrated that more musically meaningful annotations could be obtained by tapping a stable pulse around which the timing changes deviate. However, this level of tapping required extensive musical training (beyond the level of the authors) and it provides strong evidence towards rejecting beat tracking for musical pieces of this nature. For more realistic advances in beat tracking, we propose investigating techniques for music with properties that do not pose considerable difficulties for humans, such as pieces characterized by ternary meter, slow tempo, soft instrument onsets or strong voices with quiet accompaniment.
One limitation of our approach may be the use of short song excerpts for the listening test. This was done to make the listening test as manageable as possible for a wide range of participants. Nevertheless, to obtain a greater understanding of subjective ratings for longer musical excerpts and a better understanding of perceptual difficulty in beat perception it is important to conduct more sophisticated subjective listening experiments.
Furthermore, future work in beat tracking over songs with highly predominant vocals, has to be focus on automatically detect when voice suppression would have a positive effect on the beat tracking. Additionally, to extend the *Dataset Vocal* is important to consider full length stereo songs, in order to evaluate voice suppression methods in more realistic settings, because most of the voice suppression algorithms use spatial information.

Mean Mutual Agreement

In order to reliably detect difficult samples using mutual (dis-)agreement, we demonstrated that the choice of the evaluation measure is crucial, and that the Information Gain was better suited to this task than both the F-measure and AMLt evaluation methods. However, the Information Gain method appears less effective in highlighting where beat tracking algorithms strongly agree with each other. It is important to incorporate other metrics, like reliability by Degara et al. (2012), to further refine and improve results. Furthermore, we encourage to explore other evaluation methods to estimate the *MMA*, like Goto accuracy, P-score, Cemgil, CMLc, CMLt, AMLc (Davies et al., 2009a). Using our committee the performance of MaxMA, with Information Gain and Regularity, falls some way below that of the Oracle system. This suggests that there is still room to develop a more accurate selection methods using the existing algorithms.
The Mutual Agreement methods could be used over small segments in order to identify difficult sections of a song and the best beat tracker estimation could be selected for that segment, even if there are other beat trackers with better estimations for the rest of the song. It is also important to research the relation between the *MMA* value and the length of the song, because the value of the information gain measure depends on the number of beat estimations used.
The Mutual Agreement method represents an efficient way to improve diversity in existing datasets, as well as a simple technique to improve beat tracking in large non-annotated datasets. While all the directions for future work have so far been related to beat tracking, we strongly believe that, given suitable evaluation metrics, our framework based on *MMA* and *MaxMA* could be readily applied in other contexts. For instance, it may be valuable to reject the use of beat tracking as a temporal analysis component when it has detected problematic files in chord recognition or onset detection. Therefore, we encourage MIR researchers to explore *MMA* usage in problems such as onset detection, chord detection, structural segmentation, and music transcription.

Multi feature beat tracker

The proposed Multi feature beat tracker achieved better results when uses existing onset detection functions and a tracker model, contrary to recent work in the field, than designing more complex tracking models. Furthermore, beat tracking could be improved by adding a stage to analyze different metrical levels and with other onset detection functions, so we encourage the research community to work on this subject trying other onset detection functions, mixing the existing ones (Stark, 2011) or enhancing the periodicity characteristics of the audio signal with other techniques like source separation (Zapata & Gómez, 2012) or voice reduction (Section 4.2).

José Ricardo Zapata González, Barcelona, September 19, 2013.

Bibliography

Alonso, M., David, B., & Richard, G. (2004). Tempo and beat estimation of musical signals. In *Proc. International Conference on Music Information Retrieval*, pp. 158–163. Barcelona.

Alonso, M., Richard, G., & David, B. (2007). Accurate tempo estimation based on harmonic + noise decomposition. *EURASIP Journal on Advances in Signal Processing*, *2007*(1), 161.

Aylon, E. & Wack, N. (2010). Beat detection using plp. In *Music Information Retrieval Evaluation eXchange (MIREX)*.

Barabasa, C., Jafari, M., & Plumbley, M. D. (2012). A robust method for S1/S2 heart sounds detection without ecg reference based on music beat tracking. In *10th International Symposium on Electronics and Telecommunications*, pp. 307–310. IEEE.

Bello, J., Daudet, L., Abdallah, S., Duxbury, C., Davies, M., & Sandler, M. B. (2005). A Tutorial on Onset Detection in Music Signals. *Speech and Audio Processing, IEEE Transactions on*, *13*(5), 1035–1047.

Bello, J. & Pickens, J. (2005). A Robust Mid-level Representation for Harmonic Content in Music Signals. In *International Symposium on Music Information Retrieval, ISMIR*, pp. 304–311. London.

Bello, J. P. (2003). *Towards the Automated Analysis of Simple Polyphonic Music: A Knowledge-based Approach*. Ph.D. thesis, London.

Bello, J. P. (2007). Audio-based Cover Song Retrieval using Approximate Chord Sequences: Testing Shifts, Gaps, Swaps and Beats. In *Proceedings of the 8th International Conference on Music Information Retrieval*, pp. 239–244.

Berry, W. (1987). *Structural functions in music*. Dover, New York.

Bertin-Mahieux, T., Ellis, D. P. W., Whitman, B., & Lamere, P. (2011). The Million Song Dataset. In *12th International Conference on Music Information Retrieval (ISMIR)*.

Bilmes, J. (1993). *Timing is of the essence: Perceptual and computational techniques for representing, learning, and reproducing expressive timing in percussive rhythm*. Ph.D. thesis, Massachusetts Institute of Technology.

Böck, S., Krebs, F., & Schedl, M. (2012). Evaluating the online capabilities of onset detection methods. In *13th International Society for Music Information Retrieval Conference (ISMIR 2012)*, Ismir, pp. 49–54. Porto.

Böck, S. & Schedl, M. (2011). Enhanced Beat Tracking with Context-Aware Neural Networks. In *Proceedings of the 14th International Conference on Digital Audio Effects (DAFx-11)*, pp. 135–139.

Bogdanov, D., Wack, N., Gómez, E., Gulati, S., Herrera, P., Mayor, O., Roma, G., Salamon, J., Zapata, J., & Serra, X. (2013). ESSENTIA: an Audio Analysis Library for Music Information Retrieval. In *Proc. of the Int. Conf. on Music Information Retrieval (ISMIR)*. Curitiba.

Bonada, J. & Gouyon, F. (2006). Beatit, mtg.upf.edu, internal software.

Brossier, P. M. (2006). *Automatic annotation of musical audio for interactive systems*. phd theses, Queen Mary University of London.

Cannam, C., Landone, C., & Sandler, M. (2010). Sonic Visualiser: An Open Source Application for Viewing, Analysing, and Annotating Music Audio Files. In *Proceedings of the ACM Multimedia International Conference*, pp. 1467–1468.

Chordia, P. & Rae, A. (2009). Using source separation to improve tempo detection. In *Proc. 10th International society on Music Information Retrieval Conference (ISMIR)*, pp. 183–188. Kobe, Japan.

Clarke, E. (1999). Rhythm and timing in music. In D. Deutsch (Ed.) *The psychology of music*, pp. 473 – 500. San Diego: Academic Press, second edi edn.

Collins, N. (2006). Towards a Style-Specific Basis for Computational Beat Tracking. In *Proceedings of the 9th International Conference on Music Perception and Cognition*, pp. 461–467.

Cooper, G. & Meyer, L. B. (1960). *The rhythmic structure of music*. Chicago: University Of Chicago Press.

Dagan, I. & Engelson, S. P. (1995). Committee-Based Sampling For Training Probabilistic Classifiers. In *In Proceedings of the 12th International Conference on Machine Learning*, pp. 150–157.

Davies, M., Brossier, P., & Plumbley, M. (2005). Beat tracking towards automatic musical accompaniment. In *Proceedings of the Audio Engineering Society 118th convention, Barcelona, Spain*.

Davies, M. E. P. (2007). *Towards Automatic Rhythmic Accompaniment*. Ph.D. thesis, Queen Mary University of London.

Davies, M. E. P., Degara, N., & Plumbley, M. (2009a). Evaluation methods for musical audio beat tracking algorithms. Tech. Rep. October, C4DM-TR-09-06, Queen Mary University of London, Centre for Digital Music.

Davies, M. E. P., Degara, N., & Plumbley, M. D. (2011). Measuring the performance of beat tracking algorithms using a beat error histogram. *IEEE Signal Processing Letters*, *18*(3), 157–160.

Davies, M. E. P. & Plumbley, M. D. (2005a). Beat tracking with a two state model. In *Proceedings of IEEE International Conference on Acoustics Speech and Signal Processing ICASSP*, vol. 3, pp. iii/241–244.

Davies, M. E. P. & Plumbley, M. D. (2005b). Comparing mid-level representations for audio based beat tracking. In *Proceedings of the DMRN Summer Conference*, pp. 23–24. Glasgow, Scotland.

Davies, M. E. P. & Plumbley, M. D. (2007). Context-dependent beat tracking of musical audio. *IEEE Transactions on Audio, Speech, and Language Processing*, *15*(3), 1009–1020.

Davies, M. E. P., Plumbley, M. M. D., & Eck, D. (2009b). Towards a musical beat emphasis function. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 61–64. New Paltz, NY: IEEE.

Degara, N. (2011). *Signal Processing Methods for Analyzing the Temporal Structure of Music Exploiting Rhythmic Knowledge*. Ph.D. thesis, University of Vigo, Spain.

Degara, N., Rua, E. A., Pena, A., Torres-guijarro, S., Davies, M. E. P., Plumbley, M. D., & Argones, E. (2012). Reliability-Informed Beat Tracking of Musical Signals. *IEEE Transactions on Audio, Speech and Language Processing*, *20*(1), 290–301.

Desain, P. & Honing, H. (1999). Computational models of beat induction: The rule-based approach. *Journal of New Music Research*, *28*(1), 29 – 42.

Dixon, S. (1997). Beat Induction and Rhythm Recognition. In *The Australian Joint Conference on Artificial Intelligence*, pp. 311—-320. Perth, Australia.

Dixon, S. (2001). Automatic Extraction of Tempo and Beat from Expressive Performances. *Journal of New Music Research*, *30*(1), 39–58.

Dixon, S. (2007). Evaluation of the Audio Beat Tracking System BeatRoot. *Journal of New Music Research*, *36*(1), 39–50.

Dixon, S. & Pampalk, E. (2003). Classification of dance music by periodicity patterns. In *Proc. International Conference on Music Information Retrieval*, pp. 159 – 165.

Downie, J. S. (2008). The music information retrieval evaluation exchange (2005-2007): A window into music information retrieval research. *Acoustical Science and Technology*, *29*(4), 247–255.

Drake, C., Gros, L., & Penel, A. (1999). How fast is that music? The relation between physical and perceived tempo. In *Int. Conf. on Music Perception and Cognition*, pp. 190 – 203. Seoul National University Press.

Durrieu, J.-L., David, B., & Richard, G. (2011). A Musically Motivated Mid-Level Representation for Pitch Estimation and Musical Audio Source Separation. *IEEE Journal of Selected Topics in Signal Processing*, *5*(6), 1180–1191.

Duxbury, C., Bello, J., Davies, M., & Sandler, M. (2003). Complex domain onset detection for musical signals. In *Proc. of the 6th Conference on Digital Audio Effects (DAFx)*, 1. London, UK.

Eck, D. & Casagrande, N. (2005). Finding meter in music using an autocorrelation phase matrix and shannon entropy. In *6th International Conference on Music Information Retrieval (ISMIR)*, vol. 300, pp. 504–509. Citeseer.

Ellis, D. (2007). Beat Tracking by Dynamic Programming. *Journal of New Music Research*, *36*(1), 51,60.

Eronen, A. & Klapuri, A. (2010). Music tempo estimation with k-nn regression. *IEEE Transactions on Audio, Speech and Language Processing*, *18*(1), 50–57.

Gainza, M. & Coyle, E. (2011). Tempo Detection Using a Hybrid Multiband Approach. *Audio, Speech, and Language Processing, IEEE Transactions on*, *19*(1), 57–68.

Gillick, L. & Cox, S. (1989). Some statistical issues in the comparison of speech recognition algorithms. In *Acoustics, Speech, and Signal Processing*, vol. 54, pp. 2–5. IEEE.

Gkiokas, A., Katsouros, V., & Carayannis, G. (2011). ILSP Audio Beat Tracking Algorithm for MIREX 2011. In *6th Music Information Retrieval Evaluation eXchange (MIREX)*. Miami.

Gkiokas, A., Katsouros, V., Carayannis, G., & Processing, S. (2010). Tempo Induction Using Filterbank Analysis and Tonal Features. In *Proc. 11th International Society on Music Information Retrieval Conference (ISMIR)*, pp. 555–558.

Gkiokas, A., Katsouros, V., Carayannis, G., & Stajylakis, T. (2012). Music tempo estimation and beat tracking by applying source separation and metrical relations. In *proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 421–424. Kyoto.

Gómez, E. (2006). *Tonal description of music audio signals*. Ph.D. thesis, Universitat Pompeu Fabra, Barcelona.

Gómez, E. & Bonada, J. (2005). Tonality visualization of polyphonic audio. In *International Computer Music Conference*.

Gómez, E., Cañadas, F., Salamon, J., Bonada, J., Vera, P., & Cabañas, P. (2012). Predominant Fundamental Frequency Estimation vs Singing Voice Separation for the Automatic Transcription of Accompanied Flamenco Singing. In *13th International Society for Music Information Retrieval Conference (ISMIR 2012)*. Porto.

Goto, M. (2001). An audio-based real-time beat tracking system for music with or without drum-sounds. *Journal of New Music Research*, *30*(2), 159–171.

Goto, M. & Muraoka, Y. (1994). A beat tracking system for acoustic signals of music. In *the Second ACM Intl. Conf. on Multimedia*, pp. 365—-372.

Gouyon, F. (2005). *A Computational Approach to Rhythm Description*. Ph.D. thesis, Music Technology Group, Universitat Pompeu Fabra, Audio Visual Institute.

Gouyon, F. & Dixon, S. (2005). A Review of Automatic Rhythm Description Systems. *Computer Music Journal*, *29*(1), 34–54.

Gouyon, F., Dixon, S., & Widmer, G. (2007). Evaluating low-level features for beat classification and tracking. In *IEEE International Conference on Acoustics, Speech, and Signal Processing. ICASSP*, vol. IV, pp. 1309–1312.

Gouyon, F., Klapuri, A. P., Dixon, S., Alonso, M., Tzanetakis, G., Uhle, C., & Cano, P. (2006). An experimental comparison of audio tempo induction algorithms. *Audio, Speech, and Language Processing, IEEE Transactions on*, *14*(5), 1832–1844.

Grosche, P. & Müller, M. (2009). A Mid-Level Representation For Capturing Dominant Tempo and Pulse Information In Music Recordings. *10th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 189–194.

Grosche, P., Muller, M., Sapp, C. S., & Meinard, M. (2010). What Makes Beat Tracking Difficult? A Case Study on Chopin Mazurkas. In *Proceedings of the 11th International Society for Music Information Retrieval Conference*, pp. 649–654.

Hainsworth, S. & Macleod, M. (2003). Onset Detection in Musical Audio Signals. In *International Computer Music Conference (ICMC)*, pp. 136–166. Singapore.

Hainsworth, S. W. (2004). *Techniques for the Automated Analysis of Musical Audio*. Ph.d theses, Cambridge University.

Hainsworth, S. W. & Macleod, M. (2004). Particle Filtering Applied to Musical Tempo Tracking. *Journal of Advances in Signal Processing*, *15*, 2385–2395.

Handel, S. (1989). *Listening: An Introduction to the Perception of Auditory Events*. Cambridge MA: MIT Press.

Hockman, J. A., Bello, J. P., Davies, M. E. P., & Plumbley, M. D. (2008). Automated Rhythmic Transformation of Musical Audio. In *11th International Conference on Digital Audio Effects (DAFx-08)*, pp. 177–180. Espoo, Finland.

Holzapfel, A., Davies, M. E. P., Zapata, J. R., Oliveira, J. L., & Gouyon, F. (2012a). On the automatic identification of difficult examples for beat tracking: towards building new evaluation datasets. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 89–92. Kyoto.

Holzapfel, A., Davies, M. E. P., Zapata, J. R., Oliveira, J. L., & Gouyon, F. (2012b). Selective sampling for beat tracking evaluation. *IEEE Transactions on Audio, Speech and Language Processing*, *20*(9), 2539–2548.

Holzapfel, A. & Stylianou, Y. (2008). Beat tracking using group delay based onset detection. In *Proc. of ISMIR - International Conference on Music Information Retrieval*, pp. 653–658. Philadelphia.

Holzapfel, A. & Stylianou, Y. (2010). Parataxis: Morphological similarity in traditional music. In *Proceedings of the 11th International Society for Music Information Retrieval Conference*, pp. 453–458.

Khadkevich, M., Fillon, T., Richard, G., & Omologo, M. (2012). A probabilistic approach to simultaneous extraction of beats and downbeats. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 445–448. IEEE.

Klapuri, A. P., Eronen, A. J., & Astola, J. T. J. (2006). Analysis of the meter of acoustic musical signals. *IEEE Transactions on Audio, Speech, and Language Processing*, *14*(1), 342–355.

Krebs, F. & Widmer, G. (2012). MIREX 2012 Audio Beat Tracking Evaluation: Beat.e. In *Music Information Retrieval Evaluation eXchange (MIREX)*. Porto.

Lapidaki, E. (1996). *Consistency of tempo judgments as a measure of time experience in music listening*. Ph.D. thesis, Northwestern University.

Laroche, J. (2003). Efficient Tempo and Beat Tracking in Audio Recordings. *Journal of the Audio Engineering Society*, *51*(4), 226–233.

Lartillot, O. (2010). Mirtempo: tempo estimation through advanced frame-by-frame peaks tracking. In *Music Information Retrieval Evaluation eXchange (MIREX)*.

Lee, T.-C. (2010). MIREX 2010 Audio Beat Tracking Program. In *Music Information Retrieval Evaluation eXchange (MIREX)*.

Lerdahl, F. & Jackendoff, R. (1983). *A Generative Theory of Tonal Music*. Cambridge, MA: MIT Press.

Levy, M. & Sandler, M. B. (2008). Structural Segmentation of Musical Audio by Constrained Clustering. *IEEE Transactions on Audio, Speech and Language Processing*, *16*(2), 318–326.

Liutkus, A., Rafii, Z., Badeau, R., Pardo, B., & Richard, G. (2012). Adaptive filtering for music/voice separation exploiting the repeating musical structure. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 53–56. IEEE.

Malcangi, M. (2005). Source separation and beat tracking: A system approach to the development of a robust audio-to-score system. In U. Wiil (Ed.) *Computer Music Modeling and Retrieval*, *Lecture Notes in Computer Science*, vol. 3310, pp. 71–82. Springer Berlin Heidelberg.

Mandel, M. I., Poliner, G. E., & Ellis, D. P. W. (2006). Support vector machine active learning for music retrieval. *Multimedia systems*, *12*(1), 1–11.

Marchini, M. & Purwins, H. (2011). Unsupervised analysis and generation of audio percussion sequences. In *Exploring Music Contents*, pp. 205–218. Berlin, Heidelberg: Springer.

Marxer, R., Janer, J., & Bonada, J. (2012). Low-Latency Instrument Separation in Polyphonic Audio Using Timbre Models. In *Latent Variable Analysis and Signal Separation*, pp. 314 – 321. Tel Aviv, Israel: Springer Berlin / Heidelberg.

Masataka Goto, S. H., Goto, M., & Hayamizu, S. (1999). A Real-time Music Scene Description System: Detecting Melody and Bass Lines in Audio Signals. In *IJCAI-99 Workshop on Computational Auditory Scene Analysis*, pp. 31–40. Stockholm: International Joint Conference on Artificial Intelligence.

Masri, P. (1996). *Computer modelling of sound for transformation and synthesis of musical signal*. Ph.D. thesis, University of Bristol, Bristol, UK.

Mata-Campos, R., Rodriguez-Serrano, F., Vera-Candeas, P., Carabias-Orti, J., & Canadas-Quesada, F. (2010). Beat tracking improved by am sinusoidal modeled onsets - mirex 2010. In *Music Information Retrieval Evaluation eXchange (MIREX)*, 1.

Mauch, M., Noland, K., & Dixon, S. (2009). Using Musical Structure to Enhance Automatic Chord Transcription. In *Proceedings of the 10th International Society for Music Information Retrieval Conference*, pp. 231–236.

McKinney, M. F., Moelants, D., Davies, M. E. P., & Klapuri, A. (2007). Evaluation of Audio Beat Tracking and Music Tempo Extraction Algorithms. *Journal of New Music Research*, *36*(1), 1–16.

Melville, P. & Mooney, R. J. (2004). Diverse ensembles for active learning. In *the 21st International Conference on Machine Learning*, pp. 74–81.

Müller, M., Kurth, F., Clausen, M., & Muller, M. (2005). Chroma-based statistical audio features for audio matching. In *IEEE, Workshop on Applications of Signal Processing to Audio and Acoustic.*, pp. 275–278. New Paltz, NY, USA.

Oliveira, J. L., Davies, M. E. P., Gouyon, F., & Reis, L. P. (2012). Beat Tracking for Multiple Applications: A Multi-Agent System Architecture With State Recovery. *IEEE Transactions on Audio, Speech, and Language Processing*, *20*(10), 2696–2706.

Oliveira, J. L., Gouyon, F., Martins, L. G., & Reis, L. P. (2010). IBT: A Real-Time tempo and beat tracking system. In *Proceedings of the 11th International Society for Music Information Retrieval Conference*, Ismir, pp. 291–296. Utrecht.

Ong, B. S. & Streich, S. (2008). An Efficient off-line beat tracking method for music with steady tempo. In *International Computer Music Conference (ICMC)*. Belfast.

Pampalk, E. (2006). *Computational Models of Music Similarity and their Application to Music Information Retrieval*. Ph.D. thesis, Vienna University of Technology.

Peeters, G. (2009). Beat-Tracking using a probabilistic framework and linear discriminant analysis. In *12th International Conference on Digital Audio Effect, (DAFx)*, pp. 313–320.

Peeters, G. (2010). Template-based estimation of tempo: using unsupervised or supervised learning to create better spectral templates. In *Proc. of DAFX (Graz, Austria, 2010)*, pp. 6–9.

Polotti, P. (2008). *Sound to sense, sense to sound*. Logos Verlag Berlin GmbH.

Rafii, Z. & Pardo, B. (2012). Music/Voice Separation using the Similarity Matrix. In *Proc. 13th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 583–588. Porto.

Rafii, Z. & Pardo, B. (2013). REpeating Pattern Extraction Technique (REPET): A Simple Method for Music/Voice Separation. *IEEE Transactions on Audio, Speech, and Language Processing*, *21*(1), 71–82.

Ravuri, S. & Ellis, D. P. (2010). Cover song detection: From high scores to general classification. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 65–68. IEEE.

Robertson, A. & Plumbley, M. D. (2007). B-Keeper: A beat-tracker for live performance. In *Int. Conf. on New Interfaces for musical expression (NIME)*, pp. 234–237. New York.

Scheirer, E. (1998). Tempo and beat analysis of acoustic musical signals. *The Journal of the Acoustical Society of America*, *103*(1), 588–601.

Scheirer, E. D. (1997). Pulse Tracking with a Pitch Tracker. In *IEEE, Workshop on Applications of Signal Processing to Audio and Acoustics*. Mohonk, NY.

Seung, H. S., Opper, M., & Sompolinsky, H. (1992). Query by committee. In *Proceedings of the 5th annual workshop on Computational learning theory*, pp. 287–294.

Stark, A. M. (2011). *Musicians and Machines: Bridging the Semantic Gap In Live Performance*. Ph.D. thesis, Queen Mary, University of london.

Stark, A. M., Davies, M. E. P., & Plumbley, M. D. (2009). Real-Time Beat-Synchronous Analysis of Musical Audio. In *12th Int. Conference on Digital Audio Effects (DAFx-09)*, m, pp. 299–304. Como, italy.

Stark, A. M., Plumbley, M. D., & Davies, M. E. P. (2007). Real-Time Beat-synchronous Audio Effects. In *International Conference on New Interfaces for Musical Expression*, pp. 344–345.

Temperley, D. & Bartlette, C. (2002). Parallelism as a Factor in Metrical Analysis. *Music Perception*, *20*(2), 117–149.

Todd, N. (1989). A computational model of rubato. *Contemporary Music Review*, *3*(1), 69–88.

Typke, R., Wiering, F., & Veltkamp, R. C. (2005). A Survey Of Music Information Retrieval Systems. In *International Symposium on Music Information Retrieval, ISMIR*, pp. 153—-160.

Tzanetakis, G. (2010). MARSYAS submissions to MIREX 2010. Utrecht.

Tzanetakis, G., Essl, G., & Cook, P. (2002). Human perception and computer extraction of musical beat strength. In *Proceedings of the 12th International Conference on Digital Audio Effect (DAFx)*, pp. 257–261.

Tzanetakis, G. G. & Cook, P. (2002). Musical genre classification of audio signals. *Audio Processing, IEEE transactions*, *10*(5), 293–302.

Uhle, C., Jan, R., Markus, C., & Jourgen, H. (2004). Low Complexity Musical Meter Estimation from Polyphonic Music. In *Audio Engineering Society Conference: 25th International Conference: Metadata for Audio*. London, UK.

Yoshii, K. (2008). *Studies on hybrid music recommendation using timbral and rhythmic features*. Ph.D. thesis, Kyoto University.

Zapata, J. R., Davies, M. E. P., & Gomez, E. (2012a). MIREX 2012: Multi Feature Beat Tracker (ZDG1 AND ZDG2). In *the Music Information Retrieval Evaluation eXchange (MIREX 2012)*. Porto.

Zapata, J. R. & Gomez, E. (2011). Combination of Audio Tempo Estimation Approaches (MIREX 2011 Submission). In *the Music Information Retrieval Evaluation eXchange (MIREX 2011)*. Miami.

Zapata, J. R. & Gómez, E. (2011). Comparative Evaluation and Combination of Audio Tempo Estimation Approaches. In *Audio Engineering Society Conference: 42nd International Conference: Semantic Audio*, pp. 198 – 207. Ilmenau: Audio Engineering Society.

Zapata, J. R. & Gómez, E. (2012). Improving Beat Tracking in the presence of highly predominant vocals using source separation techniques: Preliminary study. In *Proc. 9th International Symposium on Computer Music Modeling and Retrieval (CMMR)*, pp. 583–590. London.

Zapata, J. R. & Gomez, E. (2013). Using voice suppression algorithms to improve Beat Tracking in the presence of highly predominant vocals. In *proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Vancouver, Canada.

Zapata, J. R., Holzapfel, A., Davies, M. E. P., Oliveira, J. L., & Gouyon, F. (2012b). Assigning a confidence threshold on automatic beat annotation in large datasets. In *Proc. 13th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 157–162. Porto.